Final Project Report

Project:

Optimizing Snow Plow Routes for the City Of Bozeman

Project Sponsor:
John Van Delinder
Street Superintendent for the City of Bozeman
Bozeman, Montana

Montana State University IMSE Senior Capstone Team:
John Corbett
Matt Crocker
Brian Locke
Ty J. Show

Advisor:
Durward K. Sobek II, Ph.D.
Professor and Program Coordinator of
Industrial & Management Systems Engineering
Department of Mechanical & Industrial Engineering
Montana State University

May 4th, 2020

Management Summary

As seniors in Industrial and Management Systems Engineering at Montana State University, the MSU senior design team was tasked with completing a capstone project with the City of Bozeman Streets Department. The City of Bozeman presented two major opportunities for improvement with regards to current snow removal operations. The main arterial roads have proved difficult to fully clear before morning traffic begins, using the current routing method. The Bozeman Streets Department has experienced some challenges with new hire turnover, and operations changes that might reduce this were welcomed. The primary objective of this project was to provide the city with a set of routes for snow plow operators that would allow for more efficient snow removal. The second objective was to propose an alternative route communication method to make the plow operator's job safer and easier.

The MSU senior design team approached the project using the engineering design process taught in previous courses in their degree. The process began by establishing the needs of stakeholders including Streets Department management and plow operators, and Bozeman residents. From there the design team moved systematically through the design process, beginning with a broad conceptual design, narrowing to a system-level design, and finally creating a detailed design and documentation of the steps taken to reach the proposed solutions. Throughout the design process the senior design team considered each decision in terms of how it addressed stakeholder needs.

Both proposed solutions began with a thorough research phase to explore alternatives and understand how these design opportunities had been approached in the past. This research consisted of consultation with faculty experts, analysis of peer reviewed studies and data collection and analysis. Data acquired included historical gps data provided by the city and observational and survey data from a ridealong with plow operators. During the research stage specific design alternatives were identified. These alternatives were then ranked using an engineering design matrix and the alternative with the highest rank with regards to stakeholder needs was selected for development.

For the snow plow routing model, a mathematical modeling method known as the Traveling Salesman Problem was selected which formed the basic structure for the model used to produce the proposed routing solution. The high priority streets and intersections, the plows and all components travel time were each mathematically described by assigning them individual variables. An optimization function was formulated to minimize the time required to plow all high priority streets in the City of Bozeman. The resulting mathematical model was then processed using a CPLEX solver in the AMPL programming language. Due to the inherent complexity of the model the design team was forced to partition the city in order to produce outputs given time constraints. It is recommended that the Street Department continue the project with an engineering consultant or future capstone team to attain truly optimal routes.

For the route communication method, the design team recommends the use of dash mounted gps units with turn-by-turn audio directions. The hands free nature of this proposed solution would make plow operation safer and easier in low visibility conditions and for new hires who are unfamiliar with the route.

Table of Contents

Table of Figures

Table of Tables

1. Overall Project Overview

1.1 Background

The City of Bozeman was named after John M. Bozeman, who established the Bozeman Trail, a route that connected the gold rush territory of Montana to the Oregon Trail. The City of Bozeman was founded in 1864, and in 1883 was incorporated as a town governed by a city council. In 1922 the local town government of Bozeman transitioned to the current city manager and city commission form of government. It is home to the largest University in the State of Montana, Montana State University, which has contributed to the growing population. It is also home to the Gallatin County seat and is intersected by both state and interstate highway systems. Responsibility for roadway maintenance is split among the various governments and institutions present in Bozeman.

Bozeman, MT is centered around geographical location, (45.6797, -111.0386), which is in the northern hemisphere, and has an area of 20.11 mi² (World Population Review, 2019). Since Montana has a northern winter climate for much of the year, snow plows are utilized in various capacities for more than 4 months of the year. “The 1981-2010 Climate Normals are NCDC's latest three-decade averages of climatological variables, including temperature and precipitation ” (National Oceanic and Atmospheric Administration, 2010). [Displayed below in Figure 1]. Figure 1 llustrates the precipitation, min temperatures, average temperatures, and maximum temperatures aggregated across the years 1981-2010 by month.

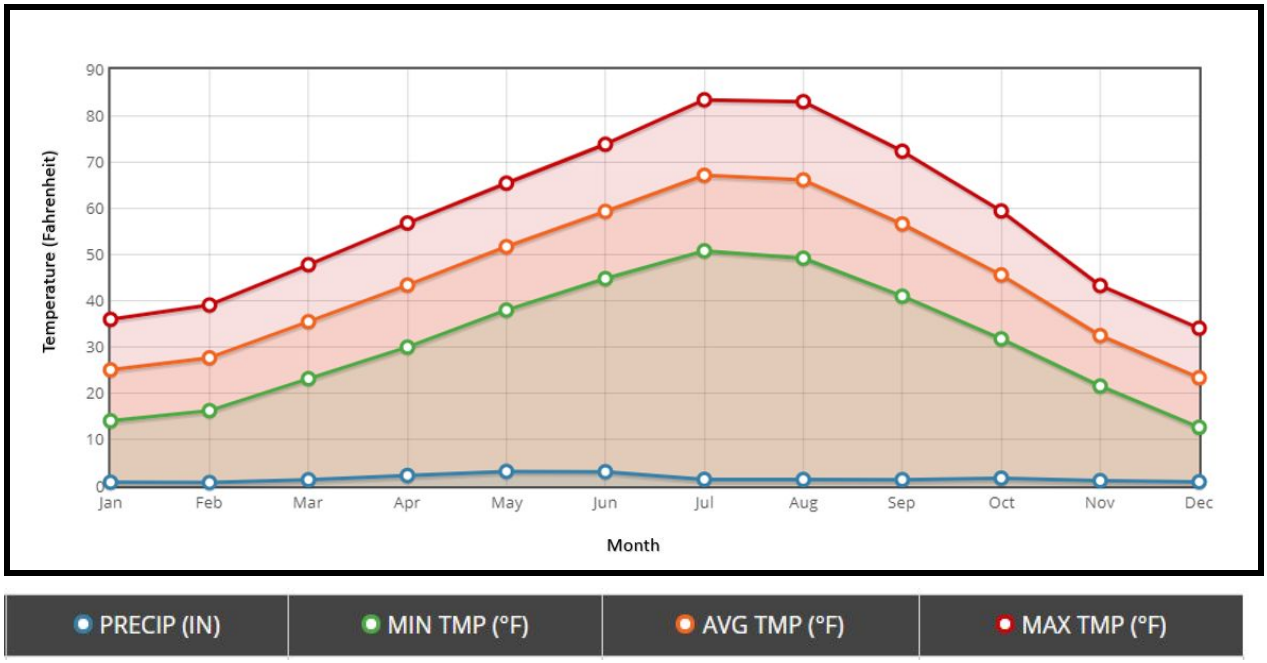


Figure 1.1 - Average Precipitation & Temperature for Bozeman, MT

Snow on roads reduces tire traction and increases stopping distances and risks of collision (Wallman, 1997). Traffic moves slowly which means roads are occupied longer and it takes more time for people to reach their destinations. Traffic due to snowy road conditions costs taxpayers money in the form of accidents and lost wages. Snow can also increase the response time of emergency services vehicles.

Cities around the world struggle with snow removal and employ a wide variety of techniques to tackle it. Most cities base their strategies on traffic and snowfall data. Most use historical data to guide routing, but some cities such as Boston, also crowd source current data from citizen reports on streets that require plowing (Nichols, 2019). Geographic information system (GIS) technology is now commonly paired with a wide variety of software to track plows, collect data, and coordinate deployment. Many cities use mathematical modeling techniques such as multiple integer programming and constraint programming, similar to those used by large shipping companies such as UPS and FedEx to optimize routes (Nichols, 2019).

1.2 Needs & Opportunity

The City of Bozeman faces a significant challenge every winter in keeping city roads clear of snow and ice. When a significant snow event occurs, the city's 8 plow trucks will start plowing as early as 3:00 a.m. in order to clear the high priority streets, ideally before 7:00 a.m. when morning traffic becomes active. Depending on the nature of the snow event, plowing may continue throughout the day. Routes have been added and expanded to match city growth of population and geographic area. The Bozeman Street Department is uncertain whether the current plow routes lead to the most efficient snow removal. Thus the first priority for this project is to develop optimized snow plow routes for the priority snow removal streets ("Snow Plowing 101", n.d.).

The City of Bozeman has undergone significant growth over the last decade. The Bozeman Real Estate Group stated on their website, "Between 2010-2018, Bozeman's population grew 4x faster than the population in the entire state of Montana, increasing by an estimated 30%. This adds up to approximately 11,196 new residents within Bozeman city limits in just 8 years. The latest data shows that Bozeman's population increased by 1,581 people in 2018 – that's more than 4 people moving to Bozeman each day" (Bozeman Real Estate Group, 2019). The Bozeman Daily Chronicle reported "Hot on the heels of a report that found Gallatin County could see 55,000 new residents by 2045, the U.S. Census Bureau last week released its most recent population estimates, which rank the Bozeman area the fastest-growing of its size in the nation" (Bozeman Daily Chronicle, 2018). "Bozeman has an estimated population of 48,532 Bozeman is the fourth most populous city in Montana" (World Population Review, 2019). These data trends, models, and assumptions, bring growth to the forefront of any future planning for the City of Bozeman. The overall population growth for Bozeman, MT between years 2011-2018 are illustrated in figure 2 below.

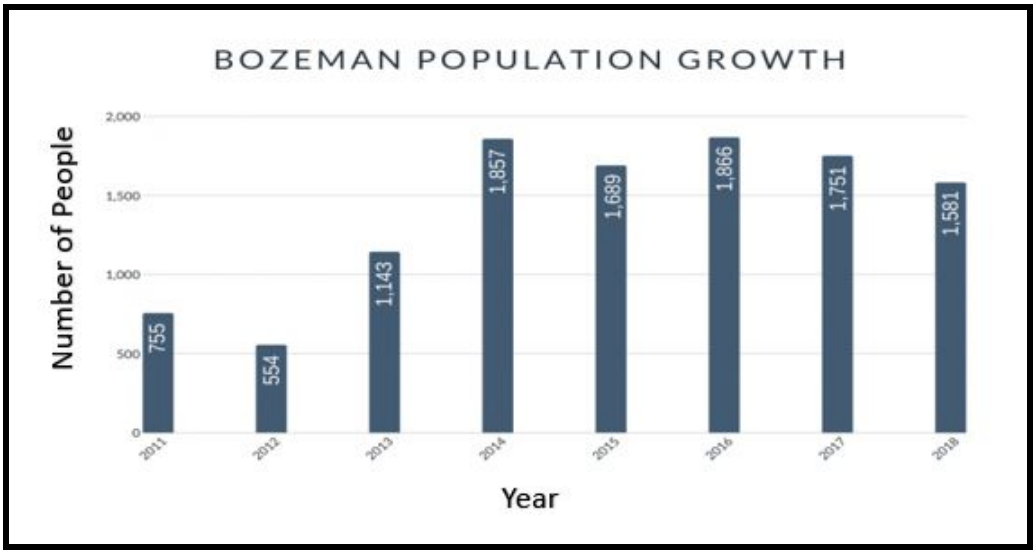


Figure 1.2 - Population growth of Bozeman, MT from 2011 through 2018

With growth, like illustrated in figure 1.2, comes the need to plan and maintain an efficient method for maintaining roadways in the City of Bozeman. There have been studies done in the past to tackle the snow plowing problems of the city, but according to the City of Bozeman Street Superintendent, John Van Delinder, it has been over a decade since a student group from Montana State University participated in a project of this nature.

With the current route distribution City of Bozeman snow plows often have trouble clearing priority roads before the morning commute begins around 7:00am. Once commuters are on the road, snow plows operate less efficiently, plowing of secondary streets is delayed further leading to accumulation of snow and ice. These problems impact citizen drivers with longer commute times, and increased risk of accidents. The city incurs costs due to increased overtime for plow drivers, and the economic impacts of traffic. As Bozeman continues to grow in size and population these costs will only increase unless plow efficiency can be improved. The City has a recurring problem of high employee turnover with their snow plow operators. Significant time and money is required to train new drivers each year. Bozeman also incurs costs when plow operators damage public and private property by crashing into things that they either do not see or are otherwise unable to avoid. There is an opportunity to reduce plow times by optimizing routes to accomplish snow removal as efficiently as possible.

1.3 Project Objectives & Constraints

The client is interested in optimized routes for the 8 plow trucks they currently own, with a goal of clearing all priority streets in 3-4 hours when a major snow event occurs. Routes and priorities for secondary residential streets should also be optimized. All proposed solutions should not add to the risk of accidents or increase employee turnover.

Project Objectives:

1. Minimize the time required for plowing high priority streets
2. Improve method of route communication to plow operators

Constraints

1. 8 plows available
2. Software should be compatible with city systems
3. Limit proposed spending

The project objectives and constraints are an instrumental component of a project because they set the stage for what is wanted by the client, as well as making sure that the project stays within designated bounds that are defined using constraints. These are hard requirements that steer solutions of the project.

1.4 Stakeholder Needs Assessment

In order to do a thorough stakeholder analysis, the senior design team met with the project sponsor and Bozeman Street Superintendent, John Van Delinder, and Zac Collins, Senior GIS Analyst. In this meeting possible stakeholders were identified. Further design team analysis uncovered the stakeholder needs that would be addressed by this project, and potential negative impacts that may arise when making changes to the current system utilized by the City of Bozeman.

John Van Delinder and Matt Workman (Customer/Project Sponsor): Street Superintendent for the City of Bozeman and Assistant Street Superintendent. The team will be reporting to Mr. Van Delinder with all Sprints and email updates on the project.

- Needs Addressed: Improved routes for the current fleet of snowplows will assist Mr. Van Delinder in his goal of snow removal from priority streets before the daily commute begins and from secondary streets throughout the winter.
- Potential Negative Impacts: If the routes developed perform worse than current routes, then time and effort spent changing the routes will have been a waste of resources.

Winter Plow Truck Drivers (End User): The team will accompany plow truck drivers on the job, to discuss how they perform their work, the common issues they face, and areas they would like to see improvement. The plow drivers need equitable routes that don't create a higher burden on some drivers than others.

- Needs Addressed: Route optimization solutions will ensure equitable distribution of work among drivers and prevent excessive use of overtime. Improved plow route communication methods such as a HUD or audio instructions which allow their focus to remain on the road, avoid accidents and reduce job stress.
- Potential Negative Impacts: Change in routine that drivers are comfortable with and potential reduction in overtime pay.

City of Bozeman (End User): The city streets must be navigable for all residents, city services and visitors throughout the winter.

- Needs Addressed: Optimized routes will ensure that the priority streets are plowed for emergency vehicles, police, schools, commuting workers and visitors. The local economy is dependent on the streets being clear, allowing people to reach businesses.
- Potential Negative Impacts: If the routes developed perform worse than current routes, there may be more accidents, economic impact due to traffic, and increase in commute times.

Taxpayers (Sponsor): The city of Bozeman's Snow Removal is funded by taxpayer dollars. The team was not given a budget for this project but since snow removal is funded by taxpayers, success would be only using a small budget.

- Needs Addressed: Optimized routes will allow the city's current fleet of snowplows to be more effective without increasing costs. Clear streets will allow residents to get where they need to go during the winter months.
- Potential Negative Impacts: Potential increase in budget. If the routes developed perform worse than current routes, there may be more accidents and increased commute times.

Summer Street Sweep Truck Drivers (End User): Optimized routes can also be used in the summer by the street sweep truck drivers.

- Needs Addressed: Solutions that ensure equitable distribution of work amongst all drivers, help in the usability of routing, and prevent excessive use of overtime.
- Potential Negative Impacts: Change in routine that drivers are comfortable with and potential reduction in overtime.

2. Methods & Engineering Design Process

2.1 Overview

During the development of this project, the MSU senior design team utilized a design process to maintain the project velocity, while integrating a systematic approach to developing a robust solution to the problem of optimizing the snow plow routes for the City of Bozeman.

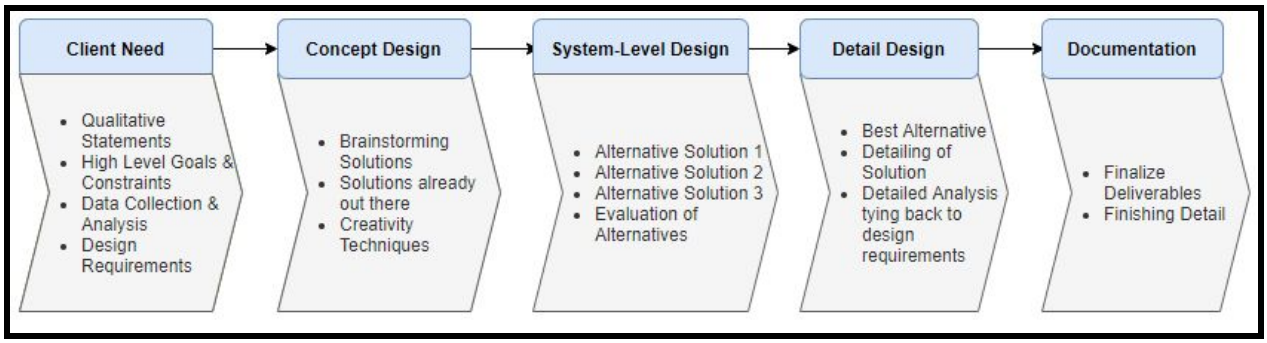


Figure 2.1 - Engineering Design Process Utilized for the City of Bozeman Project

The high level process above in figure 2.1 illustrates the steps involved with converging on a solution to optimizing the routes for snow plows. The descriptions of each step are below in section 2.2. This was a way of ensuring that the team maintained project velocity, and planned effectively so that at the end of the semester, the design team converged on a final solution, documented it, and handed it off to the client.

2.2 The Engineering Design Process Described

Client Need Description

Every project starts with a need. “Engineering and marketing models of product design and the product design process emphasize the importance of ‘Customer Needs’. Ideally, companies that develop products ought to understand customer needs so that they can develop products that will sell” (Khalid & Helander, 2004). In this case, the product isn’t being sold, but there is a buy-in component that sells the design to the client. Understanding the needs of the client is imperative in designing a product that not only meets requirements, but also fosters customer satisfaction. The client's needs are described by creating qualitative statements of high level goals, constraints, and collecting necessary data to fully understand the needs in order to design the best product in the eyes of the customer or client.

Concept Design Description

The next step in the engineering design process is concept design. “[C]oncept design is a summary and identification of the basic research. Design and research is combined early through concept design, thus the existing problems in the basic research can be found earlier and solved in a timely manner” (Chen, 2015). In this phase, the design team researches solutions that already exist, and uses brainstorming & creativity techniques to start understanding needs on a deeper level, as well as finding relevant information to converge on several design alternatives.

System-Level Design Description

System -level design is a phase in the design process that introduces different alternatives. As shown in figure 2.1 on the previous page, this level of design begins to integrate the design requirements into possible alternatives. Leveraging this knowledge to further develop alternatives was necessary in the system-level design step. “System-level design is a methodology where the engineer accounts for all the components of a system when designing the solution […] A good system-level design tool provides multiple levels of abstraction and intuitive integration between hardware and software. This makes it so that a designer can operate at a high level that takes into account the various components of the system and still produce a design that can do something in the real world” (Kodosky, 2010).

Detail Design Description

The last step in the development of a product is the detail design phase. “The detail design phase involves completing the product's design. The design team works toward completion of the specifications for the product and its subassemblies, product elements, and manufacturing processes. Like the other phases of product development, detail design is an iterative process. Design trade-offs are made as the design team learns more about the impact of design decisions on the performance, reliability, and cost of the product” (Pfeifer, 2009). Overall the detail design integrates all design requirements, while staying within the bounds of the constraints, and meets the project objectives. In this phase the model is detailed and finalized.

Documentation Description

In this last step of the overall engineering design process used, the team documents the entire project into final deliverables (i.e. reports and other project artifacts). “The Project Closing Process Group consists of those processes performed to conclude all activities across all Project Management Process Groups to formally complete the project, phase, or contractual obligations. This process group, when completed, verifies that the defined processes are completed within all of the Process Groups to close the project of phase, as appropriate, and formally establishes that the project or project phase is complete” (Aziz, 2015). For the final phase, project documentation ensures that there is an effective handoff of the project to the client so that they may begin to make decisions regarding recommendations.

3. Route Optimization Model

3.1 Overview

The goal of the route optimization model is to represent the priority snow plow routes in Bozeman, MT using a set of variables and constraints in order to mathematically determine an optimal method for snow removal. The process for developing a route optimization model began by taking a broad view of the problem being addressed and narrowing the focus incrementally, culminating in a usable solution. The design team began by working to understand the facts associated with snow removal in the city of Bozeman. Then we moved on to clarify the specific needs of the project stakeholders, Bozeman citizens and visitors, Street Superintendent John Van Delinder, and the snow plow operators. This information was then used to create problem definition, the associated objectives and the metrics that would be used to measure success of the final model.

Once the problem was defined the team acquired historical snow plowing data from the city. This data was then organized and analysed to determine what specific information would be useful in model formulation and any constraints that put on the type of modeling techniques that could be used. Next, a literature review was carried out to explore the possible modeling techniques and to understand some of the ways that this problem had been approached in the past. Through the literature review process several broad design alternatives were discovered. These design alternatives were then weighed against one another to determine which would best address the established objectives and be most feasible given the available historical data, the skills of the design team members and the project timeline.

After determining the appropriate modeling method the design team used the available data to formulate a mathematical model of the priority snow plow routes. This involved describing the priority snow plow routes by breaking them up into intersections (nodes), and the streets that connect them (arcs). Then a mathematical function was formulated in order to minimize the time required to travel along all of the required paths. Further constraints were then added to the model in order to make it as accurate as possible. Once the model was formulated it was then run through a mathematical solving program which output the final route distribution. Finally the program output was used to determine the model's validity and compared to the City's current route distribution.

3.1.1 Background of Routing Model

For the purposes of snow removal in the city of Bozeman, the streets are separated into two categories, priority and residential. The priority category consists of arterial streets and those considered critical for emergency vehicles, commuters, and commercial use. The priority streets are divided into seven separate routes, as seen in figure 3.1. A single plow is assigned to each route, except in the case of route two which covers the downtown area and is plowed by two motor graders which have the ability to plow more precisely and avoid blocking driveways, etc.. After all priority streets have been cleared, plow operators can move on to residential streets. Plowing begins as early as 3:00am and continues throughout the day.

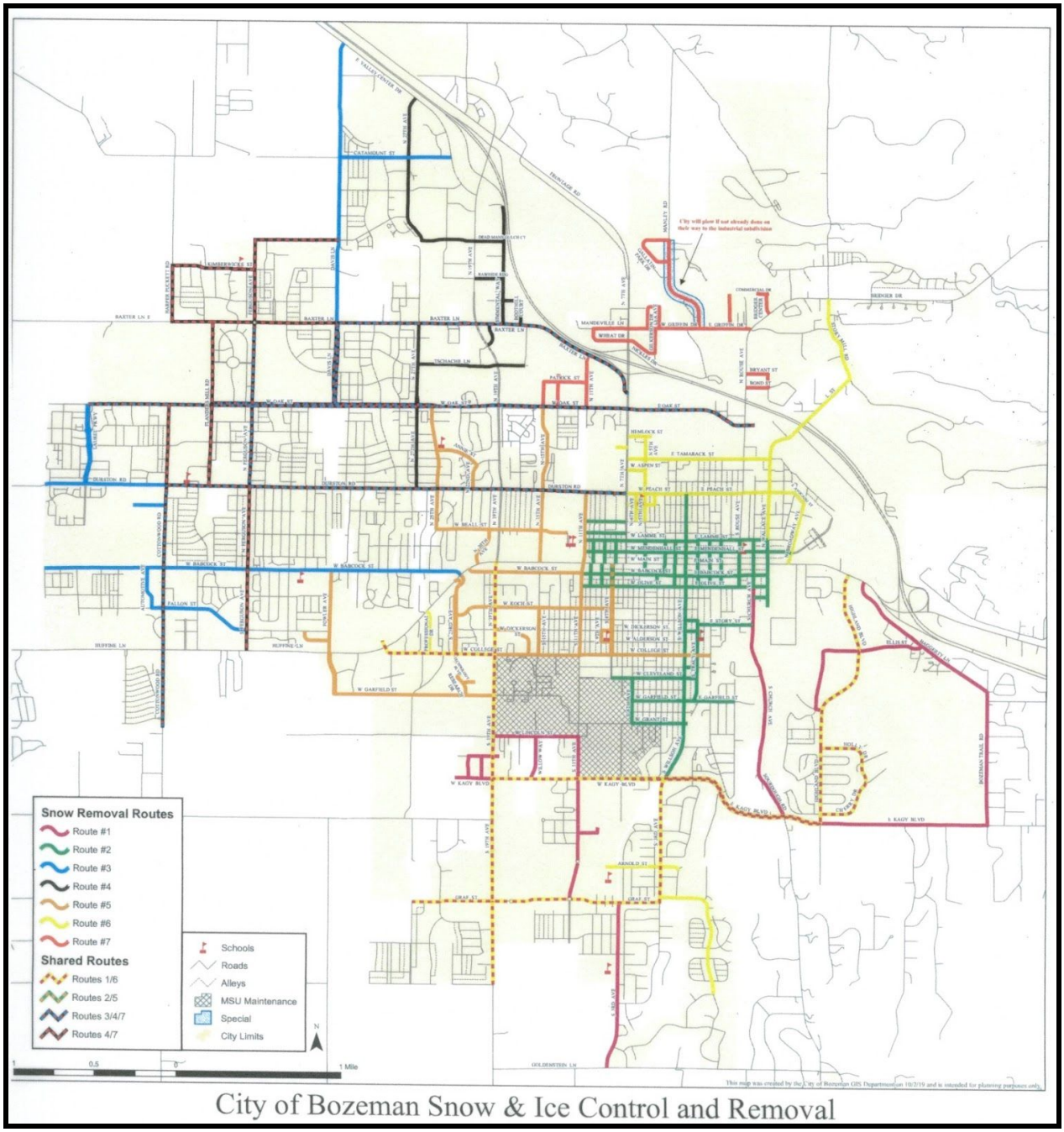


Figure 3.1 - City of Bozeman Current Priority Snow & Ice Removal Routes

3.1.2 Needs & Opportunity of Routing Model

Snow removal from priority streets should ideally be completed between 3:00am and before commuter traffic begins to increase around 7:00am. Once morning traffic begins, snow removal slows dramatically due to several factors including, the general reduction is plow drive speed, and rework due to traffic driving over partially plowed roads. With the current route distribution and the available number plows,

priority routes are rarely fully cleared in time to avoid morning traffic. According to Bozeman Street Superintendent John Van Delinder, the current route distribution was created through common sense decision making and trial and error. The routes have also expanded to accommodate the rapid growth that Bozeman has experienced rapid growth over the past decade.

3.1.3 Routing Model Problem Definition: Objective & Metrics

The City of Bozeman would benefit from a quantitative method for determining and evaluating snow plow route distribution. The method should focus on minimizing the time required to remove snow from all priority streets. Ideally a more efficient route distribution would lead to priority streets being cleared of snow within the four hour window between 3:00am and 7:00am. The routes must be feasibly completed by the five plows and two motor graders that the city currently owns. The downtown area (current route #2) must be plowed by the motor graders.

Routing Model Project Objective

- Minimize the time required for plowing high priority streets

Routing Model Metrics of Success

- Plow time or projected plow time
- Usability of plow routes
- Cost savings and increases

3.2 Data Collection & Analysis for Routing Model

City snow plows are equipped with GPS units that record data such as travel speed and geographic position of the plows. This data was provided to the design team by the city and . The data was then processed and analysed to be used as a basis for decisions relating to design of the mathematical model. In addition to the GPS data, design team members also participated in a ridealong with plow drivers and gathered observational and qualitative data.

Original Format of the Data

Nine compressed folders were received from the client that each covered a different time period: 08/01/18 to 12/31/18, 01/01/19 to 01/31/19, 02/01/19 to 02/28/19, 03/01/19 to 03/31/19, 04/01/19 to 06/30/19, 07/01/19 to 10/05/19, 10/06/19 to 11/15/19, 11/16/19 to 12/28/19, and 12/29/19 to 1/31/20. Each folder contained two metadata files and another folder inside it. Inside each of those folders there were folders for each day. Inside each day folder, there were folders for each vehicle. Inside each vehicle folder, there was an XML metadata file and a CSV file which contained the relevant data. In total there were over 4000 CSV's. Not all of the CSV's had the same order of variables, or even the same variables. However, the first 15 variables were always the following in this order: longitude, latitude, altitude, ground_speed, heading, utc_time_stamp, valid, MsgType, SatelliteCount, GsmSignalLevel, HDOP, Ignition, EventCode, Run Time Engine Start, Total Idle Time. The last variable in all the CSV's was Speed.

Cleanup Process

The first part of the cleanup process was to compile the CSV's into a single file so that they would be easier to work with. This was accomplished by writing a Python program that pulled the data from the various files and once the data was in a more manageable form, data that was clearly erroneous was deleted. There were multiple criteria used to determine the validity of each data point, and if a point failed on any criteria it was removed. Because the average altitude was 1460, it was assumed altitude was in meters (the altitude of Bozeman is approximately 1500 meters above sea level). Any point where the altitude was under 1200 meters or above 1700 meters was assumed to be erroneous and removed. The data columns ground_speed variable and Speed variable were generally the same. Any point where the two values disagreed by more than 5 mph was assumed to be erroneous and removed. The third criteria had to do with the average speed the plows would have had to travel to go from one point to the next given the time recorded at each point between each point. The great-circle distance between all points was calculated with the Haversine formula:

$$d = 2r\ arcsin(\sqrt{sin^2\left(\frac{latitude_2 - latitude_1}{2}\right) + cos\ (latitude_1)\ cos(latitude)_2 sin(\frac{longitude_2 - longitude_1}{2})}$$

The time between points was calculated from each point's timestamp, and then the average speed in-between each point and the previous point, and each point and the next point was calculated. Any point that would require the plow to be moving at 45 mph from the last point or to the next

was assumed to be erroneous and removed. After all those points were removed, the data was plotted so that problem areas could be visually identified and manually removed. A graphical summary of the data cleanup process can be seen in Figure 3.2.

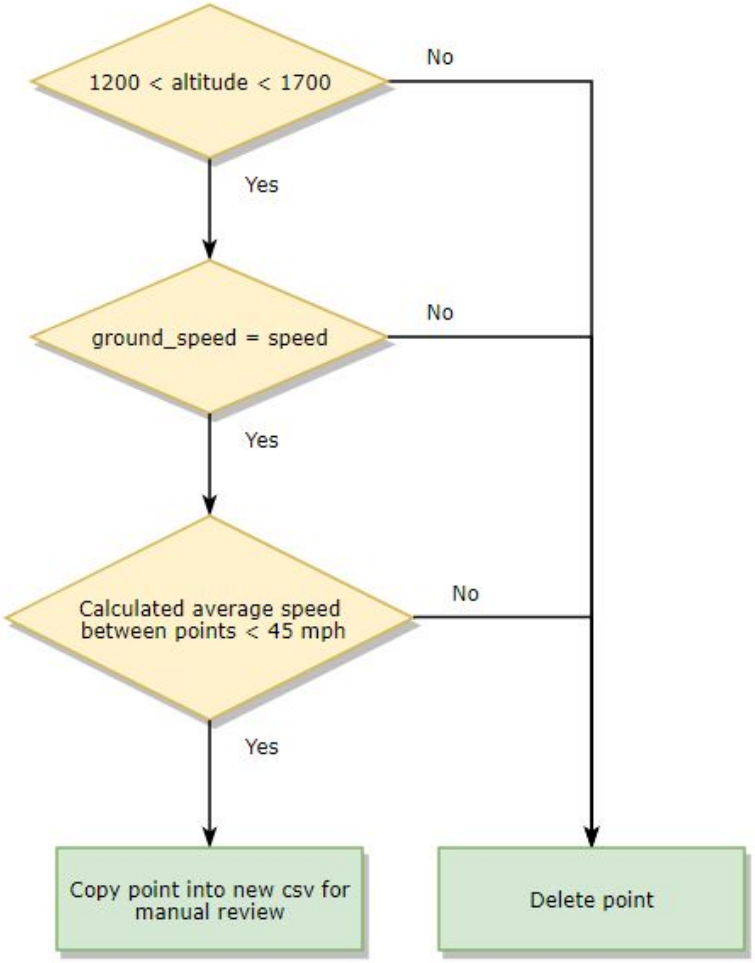


Figure 3.2 - The process used for the data cleanup

GPS Data Outputs

From the organized GPS data the design team was able to derive several pieces of information that could be used to increase the model's accuracy. The average speed while plowing was determined and the difference in speed for traveling while not plowing was negligible. The average time to turn was determined. The total time required to complete priority routes when all plows were operating was estimated.

Snow Plow Ridealong

Design team members Brian Locke and John Corbett rode with plow operators as they completed a snow removal shift from 3:00am to 8:00am. Team members gathered information by taking notes by hand on answers to prepared questions, general conversation regarding routes, and observations of the process.

This experience exposed several factors that were important for understanding the realities of snow removal. The need to avoid routes that would require reversing was indicated by both drivers. One of the drivers did extensive plowing in tandem with another driver. It became clear that it was difficult to know which lanes had already been plowed especially when it was actively snowing or when a stretch of road had been left for a long period of time while plowing other areas.

Takeaways for the Model From Ridealong

Synchronized arc routing should be incorporated if at all possible. Routes should be constructed so that turns do not require backing up. The model could include speed reduction where there is risk of covering sidewalks, affecting the cost matrix for certain arcs. Aside from those cases speed can be assumed constant even during travel outside of priority routes. Turn lanes require a separate pass of that stretch of road which can be incorporated into lane constraints. Routes should avoid long periods between passes of individual streets so that operators can more easily tell which parts of the road have been plowed. Constraints may need adjustment to avoid route suggestions involving long loops.

3.3 Similar Projects Literature Review

The design team developed a strategy for exploring alternative solutions to the snow plow routing problem that would ensure that sources contained valuable and valid information. Through this process several viable alternatives were identified. These alternatives were then further researched and summarized. In addition to the literature review, the design team consulted with faculty experts to further explore design options.

3.3.1 Strategy for Searching & Selecting Literature

The selection of relevant literature helped the design team understand how existing information may be utilized in the City of Bozeman project. Furthermore, during this research phase of the project, it was determined to be essential to understand and define the design requirements for the project. The strategy for searching for relevant information consisted of a systematic approach, utilizing academic search engines & databases to access this information. During the literature selection process the MSU senior design team wanted to use a systematic selection approach to ensure that time was not wasted reading irrelevant or invalid material. This systematic approach is shown below in Figure 3.3.

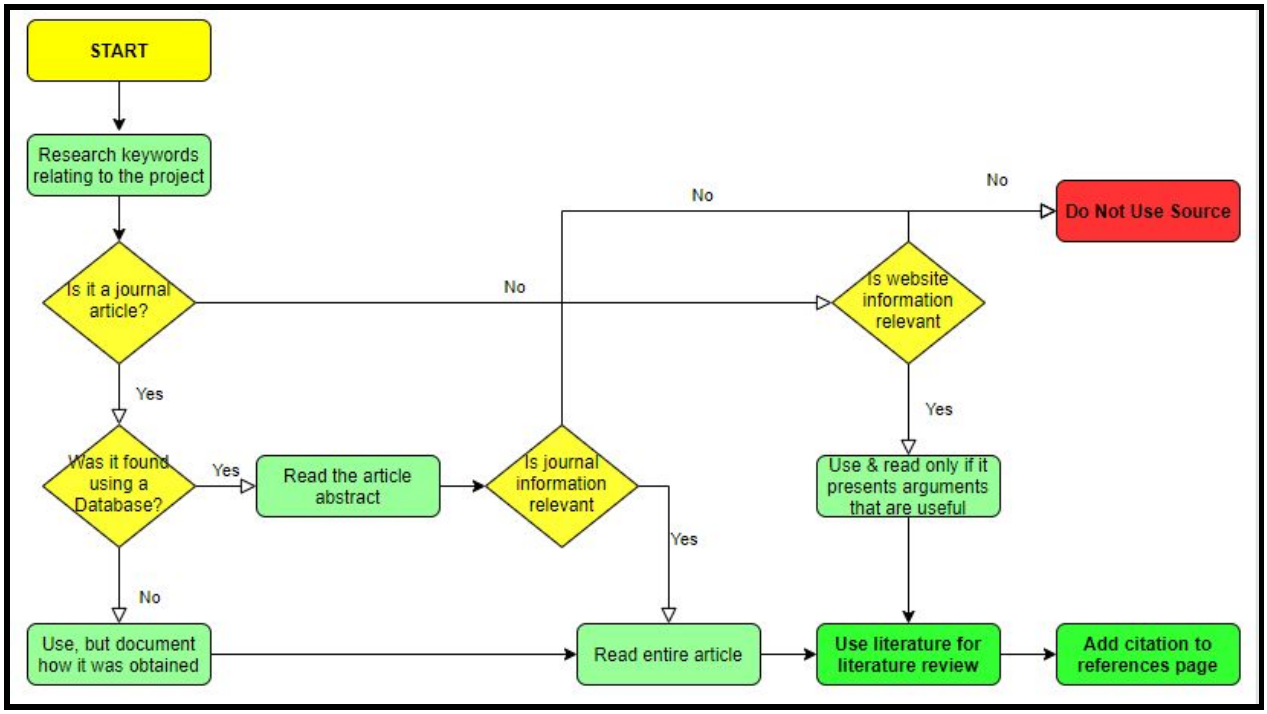


Figure 3.3 - Similar Projects Research Decision Flow chart for literature selection

This approach focused on peer-reviewed journal articles as the highest priority, followed by other sources if they contained useful information. This process was planned prior to actually conducting any research, as a method to find the relevant information from the most reliable sources. This was an internal control put on this research phase of the project in order to obtain the highest quality of information while limiting time wasted.

3.3.2 Similar Projects Literature and Snowplow Routing Models

General Chinese Postman Problem

The Chinese Postman Problem (CPP) is a standard method for finding the shortest path that covers all of the streets in a town and then returns to an origin. It is treated as a network optimization problem where all vehicles are identical, and all paths are treated as simple single lane roads. In a network diagram, roads are represented by edges and road junctions by nodes. Multiple passes must be simulated by programming a second similar path which could be in the opposite direction (Zabrodin, 2015).

Rural Postman Problem

The Rural Postman Problem (RPP) is a version of the CPP where a subset of the available routes can be designated necessary, while still allowing for travel on lower priority roads. Different versions of this method allow for the incorporation of multiple vehicles and weighted turn penalties. This method can be used to minimize overall travel time, wasted travel time, or vehicles required (Eiselt, Gendreau, Laporte, 1995).

Hierarchical Chinese Postman Problem

Another expansion on the CPP is the Hierarchical Chinese Postman Problem (HCPP). This method allows for multiple clusters of priority levels within feasible routes. Similar to RPP, HCPP has the capacity to incorporate turn penalties and restrictions. This method also allows vehicles to overlap routes when multiple passes are required (Korteweg,Volgenant, 2011).

Windy Postman Problem

The Windy Postman Problem (WPP) is a variation of the CPP which accounts for the variability of travel time along the same route depending on if the street has been plowed or not. This method is used to account for the changes due to the order in which streets are plowed (Nossack, 2016).

Synchronized Arc Routing Problem

Salazarr-Aguilar et al. introduced the synchronized arc routing problem in which the problem was formulated to plan for multiple plows going down the same road at the same time, to prevent the buildup of snow mounds (Salazar-Aguilar, 2012).

Genetic Approach to Real Time Vehicle Dispatch

A genetic heuristic for Real Time Vehicle Dispatch problems is done by finding a population of solutions for the program. A new population is generated based off of the old population and these are run through a fitness test to determine how well the solutions fit. The lower fit solutions are removed, and better fit alterations of the solution replace them. This process is continued until a final solution is reached (Mendoza, 2009).

Discussions With Faculty Experts

The design team consulted with Dr. Harris and Dr. Claudio in order to get their perspectives on potential modeling methods. Through these discussions the design team gained valuable insights on how to approach this type of optimization problem and identified one major design alternative that was not

discovered during the literature review process. This was the Traveling Salesman Model, which has a great deal of overlap with the Chinese Postman Problem identified above, but was slightly more simple and less specialised and thus had the potential to be tailored to fit this particular routing problem.

3.4 Evaluation & Selection Process of Prospective Optimization Models

During the design evaluation, three separate models were identified as displaying characteristics that apply to this project, and were evaluated using a matrix method. This evaluation matrix is described in further detail below. The three models that were evaluated were: the windy postman model, the traveling salesman model, and the synchronized arc routing model.

3.4.1 Design Requirements for Prospective Optimization Model

The establishment of design requirements is a crucial step in the engineering design process because they form the framework from which decisions are made moving forward. The design requirements were crafted to align directly with the needs of the client, as well as the overall project objectives. These design requirements are listed below.

Table 3.1 - Summary Table of Design Requirements

Requirement	Reasoning	Prioritization	Weight
Only requires inputs available from data	*The prospective model must require inputs that can be found quantitatively using the data provided by the City of Bozeman in order to make data driven changes.*	1	0.3
Can model multiple entities	*The prospective model must be able to model more than one entity because there are 7 snow plows that need to be modeled.*	2	0.2
Applicability to the situation	*The prospective model must be applicable to the situation in regards to objectives. We want to minimize the time it takes for an entity (plow) to complete a route.*	2	0.2
Complexity of the model	*The prospective model should be in a safe complexity that the MSU design team can actually model after the real-world situation, while also mathematically optimizing the snow plow routes.*	2	0.2
Compatible software	*The prospective model must be modeled, and thus must have the potential to be modeled on software that the MSU senior design team has knowledge in, or has access to help with modeling with the software available.*	3	0.05
Allows for weighting in the objective function	*The prospective model must consider equity of routes in the objective function, or allow for that elsewhere.*	3	0.05

3.4.1 Evaluation of Prospective Optimization Models

The evaluation process was systematic in nature, and provided a way to remove subjectivity when scoring the three alternatives against each other. This evaluation technique in the design process is known as an engineering decision matrix. This tool allowed the team to thoroughly examine the attributes of each alternative through a lens of objectivity that scored the alternatives, ranked them. The following image (Figure 3.4) outlines the evaluation process; yellow indicates the overall steps in the system-level design process, and the blue indicates a deeper level to accomplish those system-level design steps in terms of the City of Bozeman project.

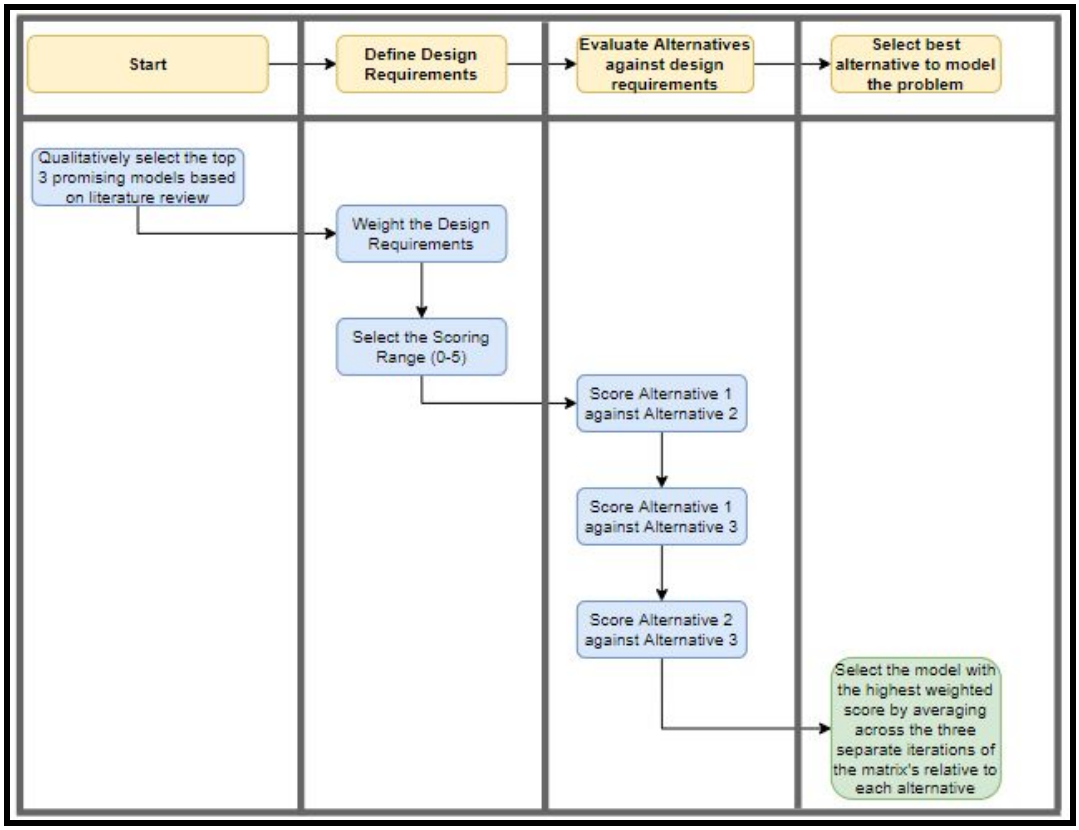


Figure 3.4 - The process for evaluation, scoring, and selection of the best modeling alternative

In order to make the evaluation process as reliable as possible, a decision matrix was utilized to evaluate the alternatives. “Decision matrix techniques are used to define attributes, weigh them, and appropriately sum the weighted attributes to give a relative ranking among design alternatives. Note that, in practice, attributes are weighted as a numeric figure based on a prescribed ranking system for individual design alternatives. In some contexts, such as design optimization, attributes are also called design objectives, which are to be maximized or minimized, or constraint functions, which must be kept within limits. In general, attributes are also referred to as design criteria or decision criteria” (Chang, 2016).

This matrix not only weighted the design requirements relative to one another, it also assigned a score to each alternative based on a 6 point likert scale (0-6) in order to evaluate alternatives. This scale is as follows: (0=completely not satisfied, 1=mostly not satisfied, 2= somewhat not satisfied, 3=somewhat satisfied, 4=mostly satisfied, 5=completely satisfied). Each cell was multiplied by the weight of that

column (design requirement), and this is why there are decimal values in each cell. The figures below, Figures C-E, illustrate each iteration evaluating each alternative relative to another. Lastly, Figure F illustrates the averages of those iterations into one summary graph where the final selection occured. Refer to Appendix B to see calculations of the scoring.

ITERATION 1 - Comparing Windy Postman & Traveling Salesman Models

TITLE: Evaluation Matrix for Alternative Modeling Techniques

		Can model multiple entities	Only requires inputs that we can get from the data	Compatible Software	Applicability to the situation	Allows for weighting in the Objective Function	Complexity to model	TOTALS
WEIGHT		**0.2**	**0.3**	**0.05**	**0.2**	**0.05**	**0.2**	**1**
Alternative	**Model**	Requirement 1	Requirement 2	Requirement 3	Requirement 4	Requirement 5	Requirement 6	**Requirement Total**
1	Windy Postman Model	0.4	1.2	0.15	1	0.05	0.6	**3.4**
2	Traveling Salesman Model	1	1.5	0.15	0.8	0.25	1	**4.7**

Figure 3.5 - Matrix iteration 1 rating windy postman relative to the traveling salesman

ITERATION 2 - Comparing Windy Postman & Synchronized Arc Routing Models

TITLE: Evaluation Matrix for Alternative Modeling Techniques

		Can model multiple entities	Only requires inputs that we can get from the data	Compatible Software	Applicability to the situation	Allows for weighting in the Objective Function	Complexity to model	TOTALS
WEIGHT		**0.2**	**0.3**	**0.05**	**0.2**	**0.05**	**0.2**	**1**
Alternative	**Model**	Requirement 1	Requirement 2	Requirement 3	Requirement 4	Requirement 5	Requirement 6	**Requirement Total**
1	Windy Postman Model	0.2	0.9	0.15	0.6	0.05	0.6	**2.5**
3	Synchronized Arc Routing Model	1	0.3	0.15	1	0.25	0.2	**2.9**

Figure 3.6 - Matrix iteration 2 rating windy postman relative to synchronized arc routing

ITERATION 3 - Traveling Salesman Models & Synchronized Arc Routing Models

TITLE: Evaluation Matrix for Alternative Modeling Techniques		Can model multiple entities	Only requires inputs that we can get from the data	Compatible Software	Applicability to the situation	Allows for weighting in the Objective Function	Complexity to model	TOTALS
WEIGHT		0.2	0.3	0.05	0.2	0.05	0.2	1
Alternative	Model	Requirement 1	Requirement 2	Requirement 3	Requirement 4	Requirement 5	Requirement 6	Requirement Total
2	Traveling Salesman Model	1	1.5	0.15	0.8	0.25	1	4.7
3	Synchronized Arc Routing Model	1	0.3	0.15	1	0.15	0.2	2.8

Figure 3.7 - Matrix iteration 3 rating traveling salesman relative to synchronized arc routing

ITERATION 4 - Summary Table of Average Scores from each iteration with weighting

TITLE: Evaluation Matrix for Alternative Modeling Techniques		Can model multiple entities	Only requires inputs that we can get from the data	Compatible Software	Applicability to the situation	Allows for weighting in the Objective Function	Complexity to model	TOTALS
WEIGHT		0.2	0.3	0.05	0.2	0.05	0.2	1
Alternative	Model	Requirement 1	Requirement 2	Requirement 3	Requirement 4	Requirement 5	Requirement 6	Requirement Total
1	Windy Postman Model	0.3	1.05	0.15	0.8	0.05	0.6	2.95
2	Traveling Salesman Model	1	1.5	0.15	0.8	0.25	1	4.7
3	Synchronized Arc Routing Model	1	0.3	0.15	1	0.2	0.2	2.85

Figure 3.8 - Final Model Matrix iteration with averages of all pairwise comparisons & highest scored

Each of the three alternatives, windy postman model, traveling salesman model, and the synchronized arc routing model, offered promising potential to apply to the City of Bozeman project. The evaluation process was detailed, tied all decisions back to the design requirements, and used methodology that allowed for selection of the best design.

Selection of Optimization Model for the City of Bozeman

After evaluating the alternatives using the design decision matrix, the MSU senior design team selected the Traveling Salesman Problem as the best alternative amongst the three. The selection process illustrated in Figure G utilized the evaluation results from the decision matrix.

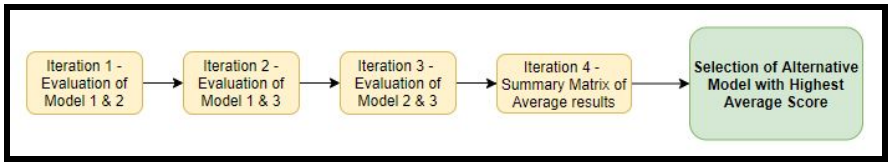


Figure 3.9 - Flow chart depicting the connection between evaluation & selection of a model

3.5 Routing Model Formulation

Traveling salesmen model selected from the evaluation matrix process was tailored specifically for the City of Bozeman's needs. The format is laid out in detail, with explanations of the various model components and a final formulation. The programming language used to solve the model is explained and the process involved in its use is detailed. Finally the validity of the model is explored based upon how well it matches historical data.

Assumptions:

In order to feasibly formulate a model that could also be solved using the resources available to the design team several assumptions were made:

- All seven plows are assumed to be operational and take no breaks
- All plows begin at the same time
- All plows travel at the same speed
- Turns always take the same amount of time

Nodes and Arcs:

Nodes represent street intersections along, or that are necessary for travel to, the priority plow routes. They are represented by i and j, where i is the starting node, j is the ending node. t_{ij} is the street, or arc, traveled between the nodes.

Variable Definitions:

i : The beginning node of an arc
j : The ending node of an arc
k : The position of an arc in a route
l : The plow identifier
t_{ijkl} : The k^{th} arc from node i to j with plow l
c_{ij} : The travel cost from node i to j
z : The total amount of travel cost on all routes

Objective Function:

The objective function represents the goal of the model and in the current iteration is designed to minimize the total amount of travel required to plow priority routes. The objective function is written as

$Min\ z = \sum_i \sum_j \sum_k \sum_l t_{ijkl} * c_{ij} + c_t * turn\ penalty\ terms$ which translates to minimizing the summation of all travel between nodes multiplied by the cost of travel between those nodes.

The map of priority routes was analyzed to determine all feasible turns that could be made by plow drivers. These intersections were meticulously recorded in a format that could be translated into AMPL

and incorporated into the model. Feasible turns consist of an origin node, a pivot node, and a destination node which were recorded in the format: [Origin, Pivot, Destination]. This resulted in a total of 950 feasible turns. Turns are then assigned a cost penalty in the form of a constant representing the average time a turn takes. The value of this constant comes from historic GPS plow location data and was roughly 30 seconds. The purpose of incorporating turn penalties into the model is to attain a more accurate estimate of completion time for the proposed routes and to incentivise the model to avoid turns in general.

Constraints:

The constraints are limits on what the model is allowed to do and how it operates. They are used to increase the model's accuracy and focus the priority of the objective function. For example, a constraint can be used to ensure the model obeys the laws of physics and does not output a solution that would require instantly jumping from one point in the city to another.

Conservation Constraints - This constraint ensures that the same number of plows that enter a node leave it. Or in other words that flow is conserved through all nodes. For each node, zero is set equal to the summation over all i's, j's and k's for all t_{ijkl}, representing arcs into the node, minus the summation over all i's, j's and k's for all t_{ijkl}, representing arcs out of the node. This is depicted in Figure 20.

Flow Constraints - This constraint ensures that each arc in each position of a route begins at the node that the arc in the previous position went to. Because the constraint is set to >= 0, when all other constraints are satisfied this constraint does not force all routes to have the maximum number of positions.

Arc Constraints - This constraint ensures that all lanes on priority streets are plowed at least once. For each set of nodes i and j with arcs connecting them, the summation over all k's & l's for all t_{ijkl}, representing arcs from node i to node j, is set greater than or equal to the number of lanes on the road moving in the direction of node i to node j. This is depicted in Figure 20.

Equity Constraints - This constraint prevents any one plow and thus any one driver from being used for over a maximum number of hours, for the purpose of avoiding creation of routes that are excessively different in length between drivers. The summation over all i's, j's, k's, l's for the product of t_{ijkl} and c_{ij} for each l is set less than or equal to a constant number of hours. The specific number of hours has yet to be determined. This is depicted in Figure 20.

Full Model Formulation

$$t_{ijkl} = \begin{cases} 0 & if\ plow\ l\ does\ not\ travel\ from\ node\ i\ to\ node\ j \\ 1 & if\ plow\ l\ does\ travel\ from\ node\ i\ to\ node\ j \end{cases}$$

$$c_{ij} = cost\ to\ travel\ from\ node\ i\ to\ node\ j$$

$$c_t = cost\ to\ turn$$

$$Min\ z = \left(\sum_i \sum_j \sum_k \sum_l t_{ijkl} * c_{ij} \right) + c_t * (turn\ penalty\ terms)$$

$$0 = \sum_i \sum_j \sum_g \sum_k t_{ijkl} - t_{jgkl} \quad \forall\ nodes \quad \forall\ l$$

$$0 \geq \sum_i \sum_j t_{ijkl} - \sum_j \sum_g t_{jgkl} \quad \forall\ j \quad \forall\ k \quad \forall\ l$$

$$\#\ of\ lanes\ that\ require\ plowing \leq \sum_k \sum_l t_{ijkl} \quad \forall\ arcs$$

$$\#\ of\ lanes\ that\ require\ plowing\ by\ graders \leq \sum_k \sum_l t_{ijk1} + t_{ijk2} \quad \forall\ arcs$$

$$\#\ of\ hours\ allowable\ per\ driver \geq \sum_i \sum_j \sum_k t_{ijkl} * c_{ij} \quad \forall\ l$$

$$1 = \sum_j t_{270j1l} \quad \forall\ l$$

Program Used to Solve Model

The programming language AMPL, and more precisely the CPLEX solver that it uses, was selected to solve the formulated model. A CPLEX solver is designed to solve large scale linear programs. The number of constraints associated with mapping a significant portion of Bozeman's streets meant that a sophisticated CPLEX solver was required.

AMPL Iterations

The design team did not have access to licenses for either AMPL or CPLEX, so the first 16 iterations of the model were run by Dr. Sean Harris. For the first few versions, a simplified version of the model was created simply to learn AMPL's syntax and CPLEX's capabilities. But even these versions revealed a problem that would plague the design team's attempts to create routes for the rest of the project: Due to

the sheer size of the network (281 nodes, >800 arcs), there were enough decision variables and constraints to make Dr. Harris's computer run out of available memory.

By version 5, progress had been made in reducing the variable count. The previous versions had defined the decision variable as a binary with 4 indices, which were the set of nodes, the set of nodes again, a counter index allowing for traveling over the same arc multiple times, and the set of plows. These sets were 281, 281, 10, and 7 elements long respectively, resulting in over 5 million decision variables. By defining the variable instead with the indices of a set of valid arcs, the counter index, and the set of plows, the number of decision variables was reduced to a mere 45 thousand. A program was coded in Python (see appendix X) to write out the set of arcs from the previously created list of nodes.

Versions 6 and 7 were used to iron out syntactical issues. On version 8 turn penalties were added. A program was coded in Python (see appendix X) to create the turn penalty terms of the objective function in AMPL's syntax. By version 9, the equity constraint functioned.

At this point, a problem was found. The routes produced were not connected. The design team consulted with Dr. Sean Harris and Dr. David Claudio to find a solution to this problem. Version 12 attempted to implement this solution, however it merely resulted in larger isolated loops instead of smaller isolated loops.

In order to solve this problem, the decision variables and formulation was changed significantly in version 14. The k index for the decision variables was changed to be the order of the arcs in the route, and so all constraints required modification. This change increased the number of variables in the model significantly, but was necessary to ensure connected routes.

On 4/26, the design team found out about the NEOS server, where CPLEX and other solvers could be run remotely. This allowed for rapid iteration of the model, however the CPLEX jobs submitted to the server were limited to 8 hours, 3 Gb of memory, and 4 threads.

All remaining versions, up to 45, were attempts to get the model to run within these limits. These can be seen summarized in table X. Tactics used to reduce the run time and memory requirements of the model include changing CPLEX options, relaxing constraints, and breaking the model into subproblems which were solved individually. While these efforts certainly reduced the quality of the solution, they were necessary to get a feasible output and were not taken lightly. Reference Appendix J to see a table of all the iterations the team went through to arrive at optimal routes, and the lessons learned along the way.

3.6 Model Validation Process

Five sets of decision variables were created to check the accuracy of the model's objective function. These represented the paths taken by plows 3028, 2682, 1806, 3769, and 2626 on 1/2/2019. The day was selected because the plows were in use for a long period of time, indicating that the data would be representative of a morning after heavy snowfall. The routes these plows took can be seen in figures 3.10, 3.11, 3.12, 3.13, and 3.14.

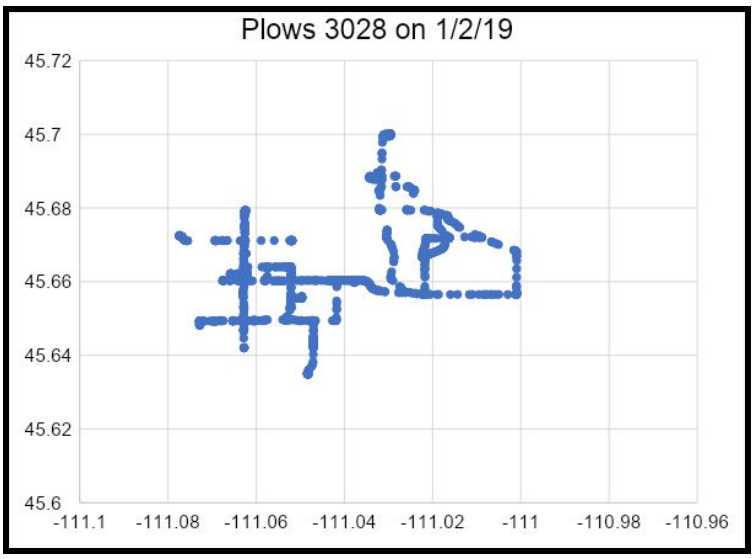


Figure 3.10 - Route taken by vehicle 3028 on 1/2/19

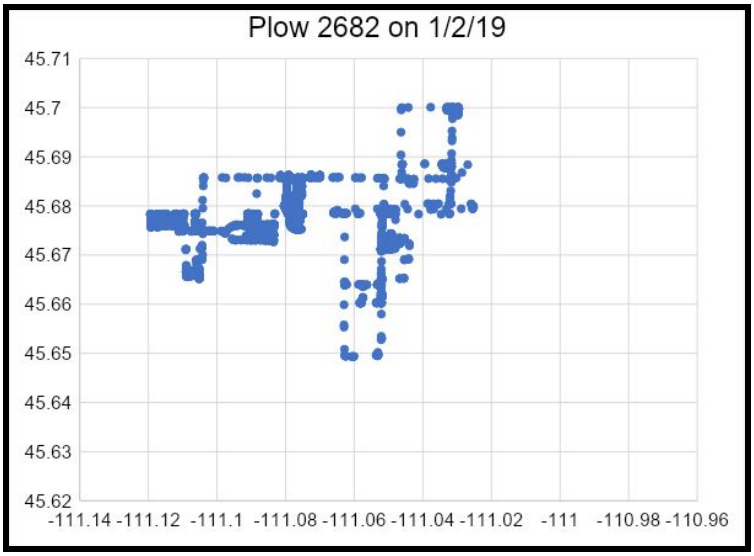


Figure 3.11 - Route taken by vehicle 2682 on 1/2/19

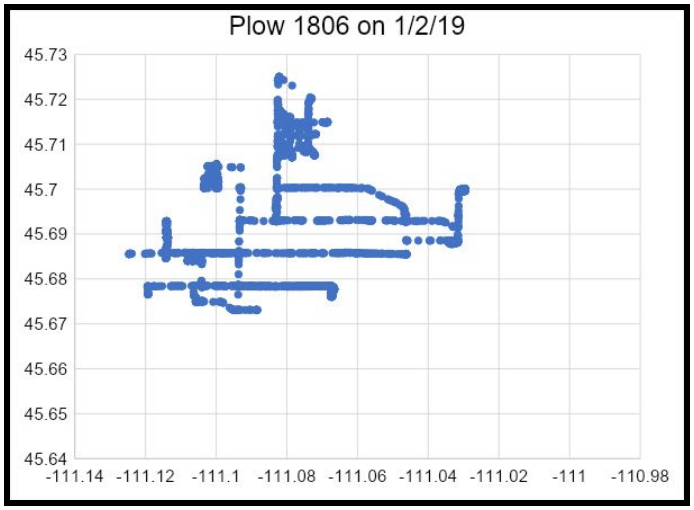


Figure 3.12 - Route taken by vehicle 1806 on 1/2/19

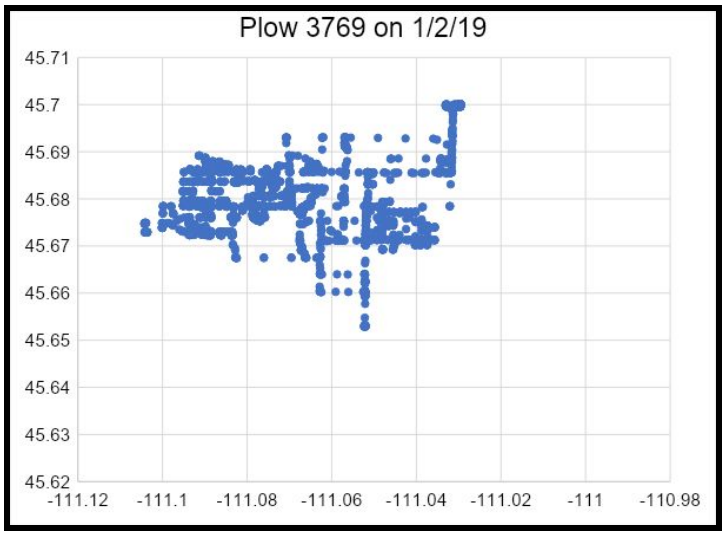


Figure 3.13 Route taken by vehicle 3769 on 1/2/19

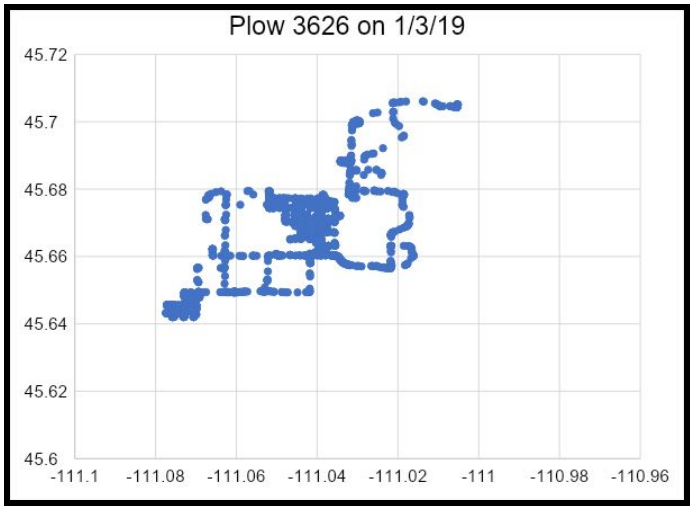


Figure 3.14 Route taken by vehicle 3626 on 1/2/19

The model was run with these variables forced to one and all other constraints removed to avoid infeasibility. The resultant times predicted by the model can be seen compared to the empirical times in table X.

Table 3.3 - Comparison of predictions made with model to recorded data

Vehicle	Route	Date	Predicted (hrs)	Actual (hrs)
3028	1	1/2/2019	6.49	6.12
2682	2	1/2/2019	8.71	8.55
1806	4	1/2/2019	6.89	7.20
3769	5	1/2/2019	7.02	7.30
3626	6	1/3/2019	7.73	7.63

With the number of assumptions that were required to feasibly model the snow plows, the average discrepancy of 15 minutes is acceptable to the design team and indicative that the model is in fact valid. Using more than five data points to validate the model would have been ideal. However, due to the time-consuming nature of this process and given schedule constraints, this was not possible.

3.7 Results and Calculated Routes

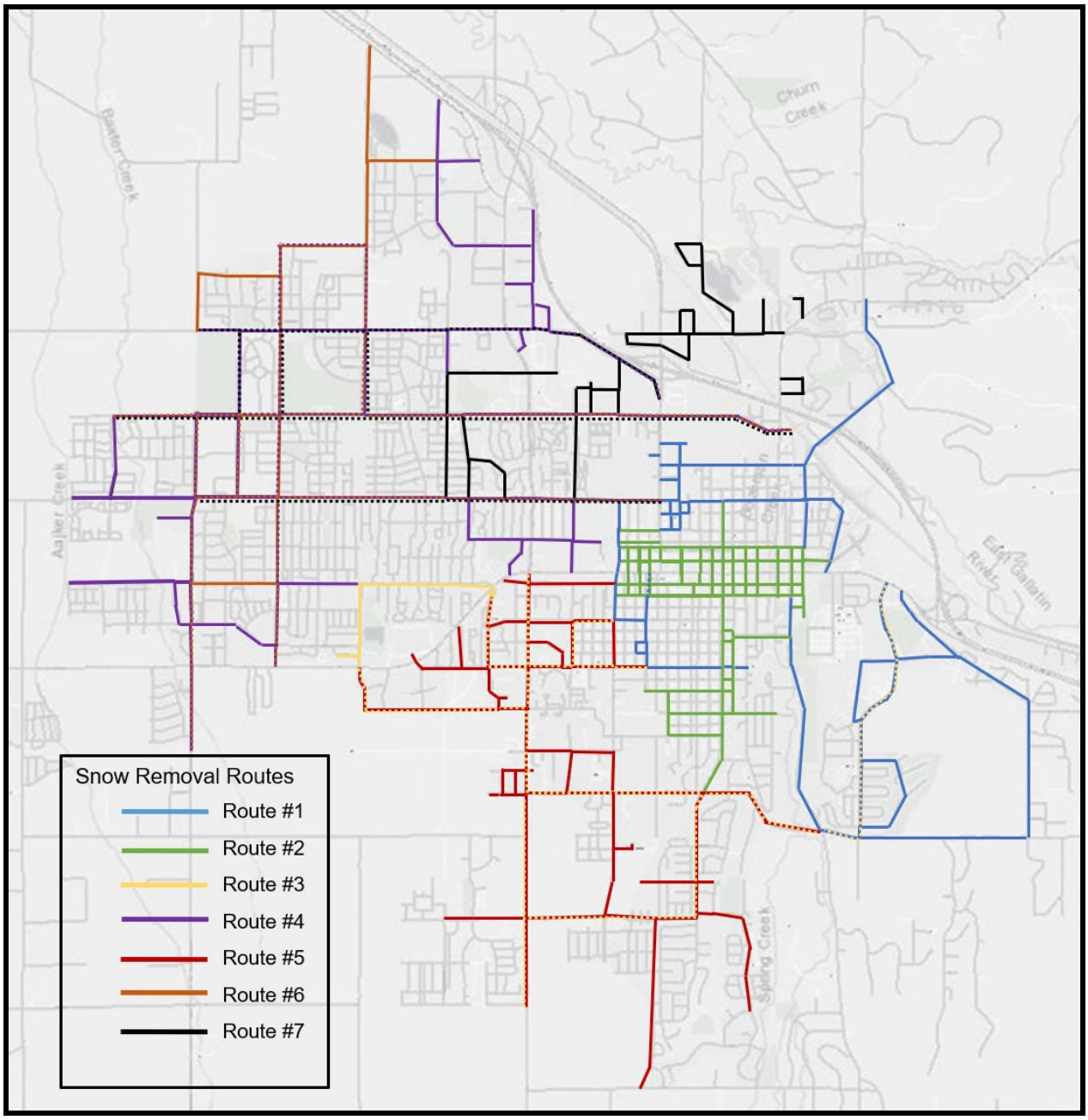


Figure 3.15 The final optimized snow plow routes for the City of Bozeman

Moving forward it is recommended that the City of Bozeman consider our proposed route changes seen in figure 3.15. Based on the value returned by the objective function for our model, the longest route should take around 4 hours and 42 minutes, while on average the routes will take 4 hours and 27 minutes. Based on the validation efforts, the design team believes that the difference from the current routes is unlikely to be because of uncertainty.

In order to confirm that our model is valid beyond the mathematical confirmation it would be necessary to compare the actual drive time of the new routes to the existing. Furthermore, it is recommended that the city work with an engineering consultant or future capstone team in order to run the full final model and continue to improve its accuracy.

4. Route Communication Method

4.1 Overview

Route communication is an integral part of the project success. The current method of route communication is a laminated paper map. There is an individual map for each route the snow plows take, and on the other side of it is a summary map of all routes put together in one. Because the MSU design team calculated new optimal routes for the City of Bozeman, it is necessary that the route communication method be analyzed. The new routes create an opportunity for a slow learning curve of the new routes. Several snow plow drivers have been clearing the same routes for years, and many times there is turnover in the positions of plow drivers. These factors led the team to a new design of route communication utilizing the robust process of engineering design. The following sections of this chapter discuss this in detail.

4.1.1 Background of Route Communication Method

In the research phase of route communication, the MSU design team looked at various existing systems used to communicate specific routes. There are many models that exist, but in order to make any communication changes value-added, the team investigated and documented current methods. In addition to that, the output of this phase was to generate overall design requirements that were carried forward in order to select a model or system that best fits the communication of snow plow routes for the City of Bozeman. The current system used to orient new drivers to a specific route is depicted in figure 4.1. The current method of communication of routes is a laminated copy of a physical map of the City of Bozeman.

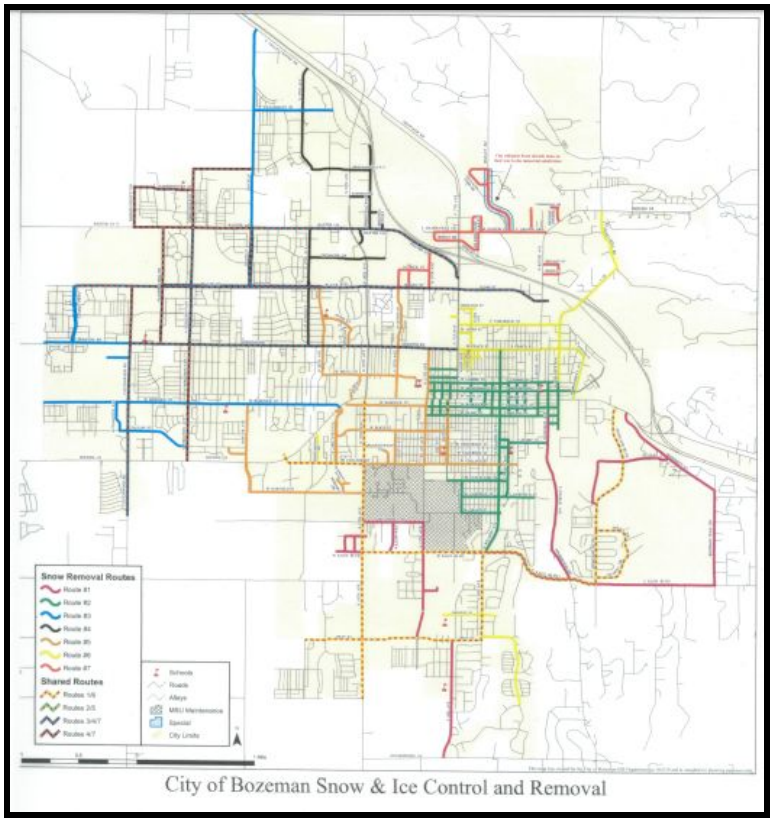


Figure 4.1 - Current State & Communication method of priority routes for the City of Bozeman

Each route is indicated with a different color, and this is the overall routing map that indicates all of the priority routes in one image. On the back of this map is another map that is specific to the route that a snow plow driver is assigned to. Since some drivers have had the same route for many years, it is imperative that if routes change, that the communication of those new routes is as effective as possible so that there are limited amounts of confusion.

4.1.2 Needs & Opportunity for Route Communication

The MSU senior design team has designed a model for the City of Bozeman that optimizes the snow plow routes. If the routes remained the same, there wouldn't be a need to change the communication method for the current drivers. Although that is true, there is still opportunity to train new snow plow drivers to learn the route because it is new to them. Most of the opportunities presented center around turnover and safety. Because there is turnover for snow plow drivers within the City of Bozeman government, it is imperative that there be an effective way to communicate the routes. Looking back at the statement of work, some plow drivers have been doing the same route for years. This presents a challenge to teach them a new route. There may be some resistance, but if there is a route communication method defined that alleviates some of the frustration of change, then the new optimized routes will be easier to switch to.

In addition to new hires being oriented to a specific route that already exists, there is also opportunity for new optimized routes, that the MSU design team generates, to demonstrate a need for a more effective communication method for the new routes. Furthermore, continuing on with safety being a part of route communication, the team wanted to investigate methods that improve the safety of the plows while transitioning to new routes, or orienting a new employee to the route. "Since automobiles are both

fabulously convenient and easy to drive, yet carry with them the constant risk of traffic accidents, attaining a high level of driving safety is of paramount importance" (ITO, 2001). Safety has been a priority and high level goal of this overall project. It is imperative that safety be integrated into any changes that may be made. Specifically, in this situation, not making the job too overwhelming on the drivers by changing their current system.

There are methods and systems that exist that integrate that safety aspect into routing communication. "Using the Heads-up Display (HUD) image position and providing visual information near the instrument panel appear to be effective in improving the ability of drivers to see road conditions outside the vehicle. For example, the HUD position leads to shorter glance time and reduced frequency of eye movement relative to other image positions. In addition, the closer the HUD image position is to the driver's forward line of sight, the greater the likelihood of detecting pedestrians" (ITO, 2001). Line of sight, and glance time are key indicators of safety when operating a vehicle. It is imperative that the eyes of the driver are on the road, as well as scanning their line of sight for any potentially dangerous threats. HUD's offer a great way to increase safety.

Global Positioning Systems (GPS) make routes easier to navigate, especially when they have a visual route on a screen, and have audio directions. "At present, GPS technology has stepped up to the more advanced level, and has several roles to perform in law enforcement, emergency services as well as in various commercial industries. Ranging from GPS vehicle tracking to GPS personal tracking and GPS kids tracking, GPS systems have addressed almost every possible need of the potential users" (GPS Tracking Solutions, 2017). There is opportunity to increase safety and help with orientation of new plow drivers. HUD's and GPS are great tools that start to delve into that opportunity to design a new, or enhance the current method of route communication. Some promising literature on these continue below.

4.1.3 Route Communication Problem Definition: Objective & Metrics

The City of Bozeman would benefit from a new routing communication method, because the city has had the same routes for a number of years. Several plow drivers have had the same route for years, and it would be an adjustment to make any routing changes. The usability of the newly optimized routes must include a way to communicate the routes to current and new snow plow drivers in a way that increases safety, as well as efficiency.

Routing Communication Method Objective

- Improve method of route communication to plow operator

Routing Model Metrics of Success

- Usability of plow routes
- Usability of route communication method

Looking through the lens of a problem definition, objectives, and metrics of success, the design team was able to integrate a solution that increases the safety and efficiency, without forgoing any of the current systems. It can be an addition to the current system of printed and laminated paper maps.

4.2 Route Communication Data Collection & Analysis

Background of Plow Driver Stakeholders

The morning of March 15th, two members of the MSU design team had the opportunity to ride with plow operators as they cleared snow from the streets of Bozeman. The plowing took place between the hours of 3:00am and 8:00am. The design team members each rode in the passenger seat of a plow for the duration of the shift, observing operations and discussing the job with the operator. The two operators who design team members rode with had differing levels of experience, one having worked with the city for 13 years, the other being in just his second year. Despite this disparity they were both clearly skilled at their job and had similar concerns when it came to what would be helpful or hindering in our routing model and communication method.

Objectives of Ridealong with Plow Drivers

The purpose of the plow driver stakeholder analysis is to gain a better understanding of the needs of the plow drivers so that they can be incorporated into the MSU design team's proposed solutions. The overarching objective of the ridealong was to gain insights with regards to the realities, priorities and challenges of the job, from the perspective of the plow drivers. The MSU design team was interested in how these insights could answer the following questions.

- How might the model need to be adjusted or focused in order to be as realistic and usable as possible?
- What aspects of route communication are the most important regarding the safety and efficiency of plow drivers?
- What challenges do plow drivers face that cannot be addressed by the model or communication method?

Findings Regarding the Model Formulation

Synchronized arc routing should be incorporated if at all possible. Routes should be constructed so that turns do not require backing up. The model could include speed reduction where there is risk of covering sidewalks, affecting the cost matrix for certain arcs. Aside from those cases speed can be assumed constant even during travel outside of priority routes. Turn lanes require a separate pass of that stretch of road which can be incorporated into lane constraints. Routes should avoid long periods between passes of individual streets so that operators can more easily tell which parts of the road have been plowed. Constraints may need adjustment to avoid route suggestions involving long loops.

Findings Regarding the Route Communication Method

Both plow operators were quite familiar with the routes and were comfortable juggling the several tasks involved in plowing. Communication methods should be geared towards accommodating new operators who do not have the same skills. Route communication methods should account for reduced visibility by informing the driver where they are, where they have been, and where they are going next. Route communication methods should allow for flexibility to accommodate improvisation by veteran plow drivers.

Additional Findings

Increased maintenance of plow mirrors, wipers and A/C would enable safer plow operation and decrease the need for periodic stops. Feedback controls such as visual indicators of when the sander was operating

should be present in all plows. Radio's could be updated for clearer communication. Additional wing plows could be used to clear roads more efficiently and would be most useful on current routes 1, 3 and 4. Cross training on different plows and routes may help operators to work together and improvise more effectively, if they choose to.

Conclusions derived from the Plow Driver Stakeholder Ridealong

The ridalongs were eye opening for the design team in terms of clarifying the difficulty of mathematically modeling real life situations. Priorities for adding detail to the model have shifted slightly with the intention of making the end product as usable as possible. Design solutions for route communication methods should take findings from the ridalong into account. Overall, this experience has provided the team with a better understanding of how the model and communication method will affect the end user.

4.3 Promising Communication Methods Literature Review

Safety is the main goal when it comes to route communication. The current model has the plow operators looking away from the road to view a map or navigating their routes off memory. Many of the plow drivers are not from the Bozeman, and new drivers need to learn their routes. The four route communication methods we have researched are Heads Up Displays, GPS Units, Smartphones with Routing Apps, and Feedback Controls on Dashboard.

4.3.1 Heads Up Displays

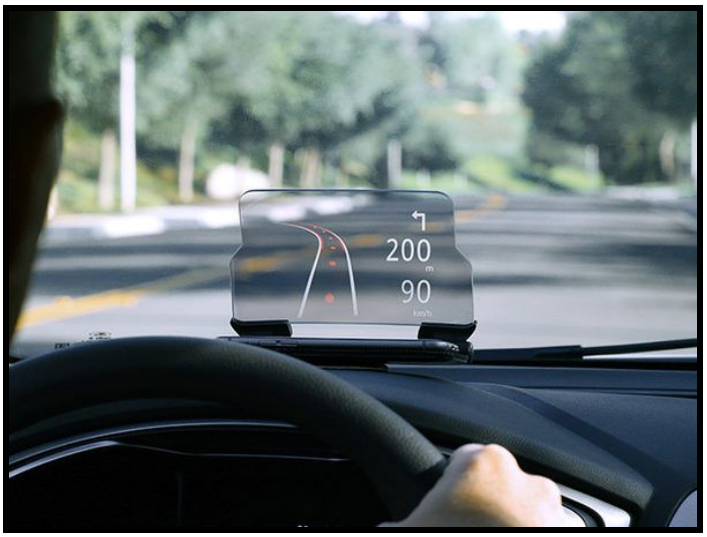


Figure 4.2 - Dash mounted HUD

https://shop.popsci.com/sales/hudway-glass-heads-up-display?gclid=CjwKCAjwguzzBRBiEiwAgU0FT9w8DDp5nIMDNAwsgFqGE8rcG_MGb8JxB1nBERWPzXOR2CYzawj7TxoCl48QAvD_BwE

Relevance Heads Up Displays

A Heads Up Display (HUD) is a transparent screen or projection that displays information without the user having to change their viewing angle. During World War II, pilots received verbal directions regarding the locations of their targets. The pilots struggled with this method of communication and needed a solution that allowed them to keep their eyes on the flight path. Heads Up Displays were developed from this problem. Other than pilots, the air force uses HUDS to provide paratroopers with on the fly directions to landing zones (Ontiveros, 2008).

A dash mounted HUD can be seen in Figure 6. For the plow drivers, a HUD would be used to display routing information such as upcoming turns and vehicle speed. Aftermarket automotive HUDs use bluetooth connections to phones. Optimized routes would be programmed into phone navigation apps for the drivers to use and these routing apps rely on cellular service to operate properly.

Benefits of Heads Up Displays

The HUD displays information in the driver's field of vision, making it safer than looking down at a phone or map for directions. A study performed by the Aviation Research Laboratory at the University of

Illinois found that, “time to respond to both near domain symbology events and far-domain helicopter-target images was consistently faster with the head-up display than with the head-down presentation” (Wickens & Ververs, 1998). Unlike the current situation where the drivers are using printed maps for the routing, a HUD system would allow them to keep their eyes on the road while receiving directions which is inherently safer than looking down at a map. The University of Illinois also found that “HUDs allow scanning the environment and scanning the superimposed instrumentation to be a matter of dividing attention between these far and near domains, respectively” (Wickens & Ververs, 1998). Audio directions can also be available with phone navigation apps. Aftermarket HUDs are universal and easily mounted to the dash of a vehicle. HUDs are easy to view in the dark; the plow drivers operate at night/early morning when visibility is poor.

Drawbacks Heads Up Displays

While HUDs are easy to obtain, they can cost anywhere from $50 to $300 per unit when purchased on sites like Amazon. Additionally, they mirror what is displayed on the phone they are connected to, they are not their own navigation device. The HUD may be distracting to some users since navigation symbols and lighting are now in their field of view. The study done by the University of Illinois found that in one scenario where a HUD was being used, “An incident was noted in a military report that excessive brightness, combined with an overload of symbology, probably resulted in the fixation on the display and distraction of the pilot. In the final report, the presence of the HUD was named as a contributing factor to the incident” (Wickens & Ververs, 1998). Each driver will need access to a phone with navigation apps/routes to utilize a HUD. A drawback of being tied to a phone is that they can lose cellular service or bluetooth connection causing failure between devices and loss of routing information.

4.3.2 GPS with Audio Directions

Figure 4.3 - GPS with Audio unit
https://images.wisegeek.com/gps-on-a-car.jpg

Relevance of GPS with Audio Directions

Global Positioning Systems (GPS) is a satellite navigation system used to determine the ground position of an object. GPS was developed by the United States Military in the 60’s and uses 32 satellites that

orbit the earth every 12 hours (FAA, 2014). An example of a dash mounted GPS unit can be seen in Figure 7 above. GPS units can be mounted to the dash or windshield of a vehicle using suction cups or adhesives. Routes would be programmed into the GPS units for the drivers to use.

Benefits of GPS with Audio Directions

Unlike the HUD, GPS is a standalone unit and does not require a phone. GPS is operated and maintained by the Department of Defense and provides users with accurate positioning information (FAA, 2014). GPS units provide both verbal and visual communication of the routes. GPS units can be mounted in a variety of different ways such as suction cups, vent mounts, or dash mounts. These options allow drivers to choose where the device is mounted. Routes are programmed into the unit and do not require cellular service to operate. Much like the HUD units, GPS units are easy to obtain. GPS units provide good visibility in dark conditions.

Drawbacks of GPS with Audio Directions

GPS units can cost anywhere between $50 to $400 per unit when purchased online. Unlike the unrestricted field of view that the HUD provides, the GPS must block the field of view or cause the user to look away from the road to view the device. Even though they don't rely on cellular networks to provide routing, connections to the satellites can fail resulting in loss of routing directions. Reasons for these failures can be signal blockage due to buildings, indoor or underground use, and signals reflected off buildings or walls (GPS Accuracy, 2017).

4.3.3 Phone with featured App/Maps

Figure 4.4 - Phone with featured map application
https://www.komando.com/wp-content/uploads/2019/07/map-apps-for-multiple-stops-.jpg

Relevance of Phone with featured App/Maps

Much like GPS units, smartphones can be used for navigation. Navigation apps for smartphones (example in Figure 8) use cell tower assisted GPS signals, called AGPS, to determine the location of the phone (Hildenbrand, 2018). These apps can be used to display optimized routes to the plow drivers in real time. Phones can be mounted to the dashboard or windshield of the vehicle using magnetic or suction cup mounts.

Benefits of Phone with featured App/Maps

The cellular GPS that smartphones use have an accuracy within a 4.9 m to the actual position of the user (GPS Accuracy, 2017). Aside from the accuracy of smartphone GPS, Smartphones can be mounted in a variety of ways. Suction cup and magnet mounts allow drivers to choose where the phone is mounted. Phone map apps provide both verbal and visual communication. Phones provide excellent visibility in dark conditions, and the apps can be downloaded on personal devices.

Drawbacks of Phone with featured App/Maps

If not all the employees own smartphones or want to use their personal phone for work, the cost of phone and cellular plan must be considered. Since the phones are mounted on the dash or windshield of the vehicle, they block or take the users eyes away from the road. In 2018, 2,841 people were killed in motor vehicle crashes involving distracted drivers (Currin, 2020). As well as blocking vision, phones receive other notifications such as text messages which can be distracting for the drivers. Phones can lose cellular service, causing loss of routing information, and navigation apps drain phone batteries quickly.

4.3.4 Original Paper Maps

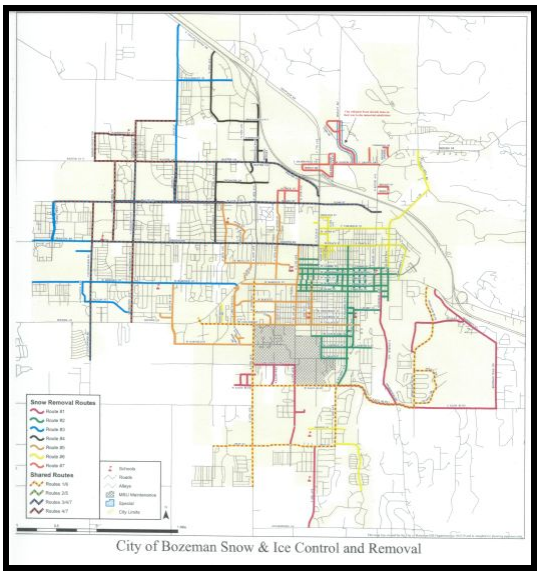


Figure 4.5 - Comprehensive original paper map of all current priority streets

Relevance of Original Maps

Currently, the City of Bozeman plow drivers operate their routes based on physical printed maps. Figure 9 is the map of all the current plow routes that the city uses. The drivers have a map of their individual route and bring it with them into the vehicle for reference. The plow drivers operate from 3:00 A.M. to 7:00 A.M., during these hours, it is dark out. Since maps do not have lights built into them like other options, the driver must use an interior light in the vehicle. The benefits and drawbacks of the Original Maps are below.

Benefits of Original Maps

Unlike both HUD's and GPS units, printed maps are tangible items that the user holds. Since the maps are printed they will always show the route. It doesn't matter if the user is in an area without cellular service, the maps cannot lose connection. The City of Bozeman is currently using printed maps for route communication, so providing the drivers with new maps would be seamless and easy to implement. The final benefit of printed maps is that they are inexpensive to create and reproduce. Printing of the maps is done in house, and would not add much cost to the city.

Drawbacks of Original Maps

The current maps the plow drivers are using are laminated 11"x17" sheets of printed paper. If they do not have their route memorized, they are required to look down at the map which takes their eyes away from the road and could lead to accidents. Much like the other heads down display options, "Any non-driving activity you engage in is a potential distraction and increases your risk of crashing" (Currin, 2020). On top of this, the maps are not possible to see in the dark so the drivers must turn an interior light on to view them. The other downside to maps is that they cannot be updated like the HUD and GPS units. When new roads are built in the city, the maps will need to be reprinted with the new updated routes.

4.4 Evaluation & Selection Process of Route Communication Method

4.4.1 Stakeholder Analysis for Route Communication

It is important to revisit the statement of work during the design process of the route communication methods portion of the City of Bozeman project, where the MSU design team has been optimizing snow plow routes. All objectives for this design phase need to tie back to an original goal or objective of the overall project. These objectives are in the statement of work. Tying all goals back to the overall project objectives is crucial in ensuring that the client gets what they expect. The original project objectives and metrics of success are below:

Project Objectives *(from statement of work)*:

- Minimize the time required for plowing high priority streets
- Similarly optimize routes for residential city roads
- Quantify the impact of snowfall on completion time
- Improve method of route communication to plow operator

Project Metrics *(from statement of work)*:

- Plow time or projected plow time
- Cost savings
- Usability of plow routes
- Usability of route communication method

The most useful project objective that can be carried into the design process of communication of routes is “Improve method of route communication to plow drivers.” The most useful metrics of success that can be carried forward in the design process of communication of routes are, “Usability of plow routes,” and “Usability of route communication method.”

For the design of the route communication method, it is also important to integrate the feedback discussed in the previous part of this report (Plow Driver Stakeholder Analysis). In addition to the feedback from the plow drivers it is important to revisit the statement of work. Particularly, the stakeholder needs analysis. In the beginning of the project, the MSU design team forecasted the stakeholders involved in this project, their needs being addressed, and the potential negative impact. Next, are the plow driver stakeholder summary from the statement of work.

Plow Driver stakeholder analysis *(from statement of work)*:

- Winter Plow Truck Drivers (End User): The team will accompany plow truck drivers on the job, to discuss how they perform their work, the common issues they face, and areas they would like to see improvement. The plow drivers need equitable routes that don’t create a higher burden on some drivers than others.

- Needs Addressed: Route optimization solutions will ensure equitable distribution of work among drivers and prevent excessive use of overtime. Improved plow route communication methods such as a HUD or audio instructions which allow their focus to remain on the road, avoid accidents and reduce job stress.

- Potential Negative Impacts: Change in routine that drivers are comfortable with and potential reduction in overtime pay.

This approach of revisiting the statement of work from the beginning of the project ensures a more robust and thorough process for integrating this information into the design of the route communication method. Specifically, all sections that follow for this part of the report further discuss the intentional integration of the statement of work into the design requirements used to evaluate alternatives.

4.4.2 Defining Design Requirements for Route Communication

This is a very crucial step in the engineering design process because design requirements are the framework from which decisions during the selection process will be managed. This is where the goals and constraints are defined, and then aligned with the design requirements that are generated. The MSU design team will carry forward the promising communication methods, develop high level goals and constraints that tie to the overall project objectives and metrics of success. After the goals and constraints for the design of communication methods are developed, the design requirements will be developed and prioritized by level of importance, assigned a weight, and utilized to score each promising method against the design requirements. It is important to tie it all back to the overall project goals and metrics because it is a guide for developing a robust solution.

Approach to Defining the Design Requirements

As part of the engineering design process, one of the very first steps is generating the design requirements.“The Design Process begins with design requirements and ends with product descriptions. The design requirements include both structural and performance aspects.” (Zeng, 1999). Furthermore, the design requirements align directly with the needs of the client, as well as the overall project objectives, metrics of success, and plow driver stakeholder analysis. First, the high level goals and constraints are generated by tying it all back to the statement of work’s overall project goals, metrics of success, and plow driver stakeholder analysis.

High Level Goals and Constraints of Route Communication Method

During this process of defining the design requirements, the MSU design team started with defining the goals and constraints of the route communication changes. These high level statements help navigate the myriad of models that currently exist. The goals indicate measurable goals that will align with the design requirements.

Route Communication Method Goals

The goals of designing a new route communication method are listed below:

- Increase the usability of snow plow route communication
- Create countermeasures for line of sight distractions due to current map method
- Decrease distractions due to route communication
- Increase amount of standard operating procedures (SOP's) for method

Route Communication Method Constraints

The constraints of designing a new route communication method are listed below:

- The countermeasures should be relatively easy to install
- The countermeasures should be low cost
- The new system should stay in line with the overall project goals
- The new system should not make the job harder or less safe

4.4.2 Design Requirements for Route Communication

To start, there are high level goals and constraints defined based on project goals and metrics of success. Then the design requirements are generated using the goals and constraints that are based on the overall project goals and metrics from the statement of work. Everything is tied together. The overall project goals to the communication method goals to the design requirements that are generated.

The design requirements were derived from an extensive process identifying factors that result in the success of this project. Overall, the models must have the following factors in order to effectively design a product that meets the clients needs:

Definitions:

1. Reduces chance of accidents - *The prospective model must demonstrate safety features, or ways to implement safety into the design. This could be anywhere from fail safe features to the positioning of any new informational interfaces. There shouldn't be an overbearing noise, lighting, flashing, or any other distractions beyond reasonable multi-tasking.*
2. Limited line of sight movement - *The prospective communication method must reduce the amount of time that the driver will have to take their eyes off the road in order to effectively navigate their route or maneuver the snow plow they are operating. The line of sight is important because the eyes should stay on the road and less on other distractions.*
3. Programmable, or ability to input different routes - *The prospective communication method must have the ability to integrate predetermined routes, or be able to be programmable with a route based on global positioning or other mapping technology. Furthermore, this is required because the new routes may be different, and there needs to be a way to input the new routes into the method or model.*
4. Limited complexity of technique (usability) - *The prospective communication method must have limited complexity because the MSU team won't actually be a part of implementation. In addition, it is not in the best interest of the snow plow drivers to overly complicate their ability to carry out their driving duties. Furthermore, there shouldn't be an added distraction of learning a complex device.*
5. Easy implementation - *The prospective communication method must demonstrate an ability to have easy installation and low cost, and educational steps to train the snow plow drivers to operate the new proposed system. Furthermore, there should be minimum physical changes to the interior of the cab to minimize electrical changes or other complex changes.*

Priority & Weighting of Design Requirements

After the design requirements were defined, the MSU design team assigned priority numbers to each design requirement, with 1 being highest priority. It was decided that both increased safety and limiting line of sight were deemed the most important with the highest priority of one. Next, the requirement of being able to input a route was the next priority with two. Finally, the requirements of easy implementation and limited complexity took the last priority with three. After the design requirements were prioritized, they were assigned a weight of importance. These weightings tie back to the prioritization which tie back to the design requirements that tie back to the goals and constraints of designing a route communication method, and finally those tie back to the overall project objectives and metrics. It's a robust and holistic approach to evaluating the alternatives.

Summary Table of Design Requirements

The table below illustrates the design requirements that will be used to determine which new system or model will align with the project goals and fall within the bounds of the constraints defined in the last section. This step is crucial in determining which model or models to select as a way of creating a more effective route communication system.

Table 4.1 - Summary Table of Design Requirements for Route Communication

Requirement	Reasoning	Priority	Weight
Limited Complexity of Techniques (Usability)	*The prospective model must have limited complexity because the MSU team won't actually be a part of implementation. In addition, it is not in the best interest of the snow plow drivers to overly complicate their ability to carry out their driving duties.*	3	0.1
Easy Implementation	*The prospective model must demonstrate an ability to have easy installation, and educational steps to train the snow plow drivers to operate the new proposed system.*	3	0.1
Programmable, or ability to input a route	*The prospective model must have the ability to integrate predetermined routes, or be able to be programmable with a route based on global positioning or other mapping technology.*	2	0.2
Reduces chance of accidents	*The prospective model must demonstrate safety features, or ways to implement safety into the design. This could be anywhere from fail safe features to the positioning of any new informational interfaces.*	1	0.3
Limited line of sight movement	*The prospective model must reduce the amount of time that the driver will have to take their eyes off the road in order to effectively navigate their route or maneuver the snow plow they are operating.*	1	0.3

4.4.3 Evaluation of Prospective Route Communication Methods

The evaluation process was systematic in nature, and provided a mostly objective way to score the four alternatives, including the original paper maps, against each other. This evaluation technique in the design process is known as an engineering decision matrix. This tool allowed the team to thoroughly look at the attributes of each alternative through a lens of objectivity that scored the alternatives, ranked them, and selected the best one. The following figure 4.6 outlines the evaluation process; yellow indicates the overall steps in the system-level design process, and the blue indicates a deeper level to accomplish those system-level design steps in terms of the City of Bozeman project.

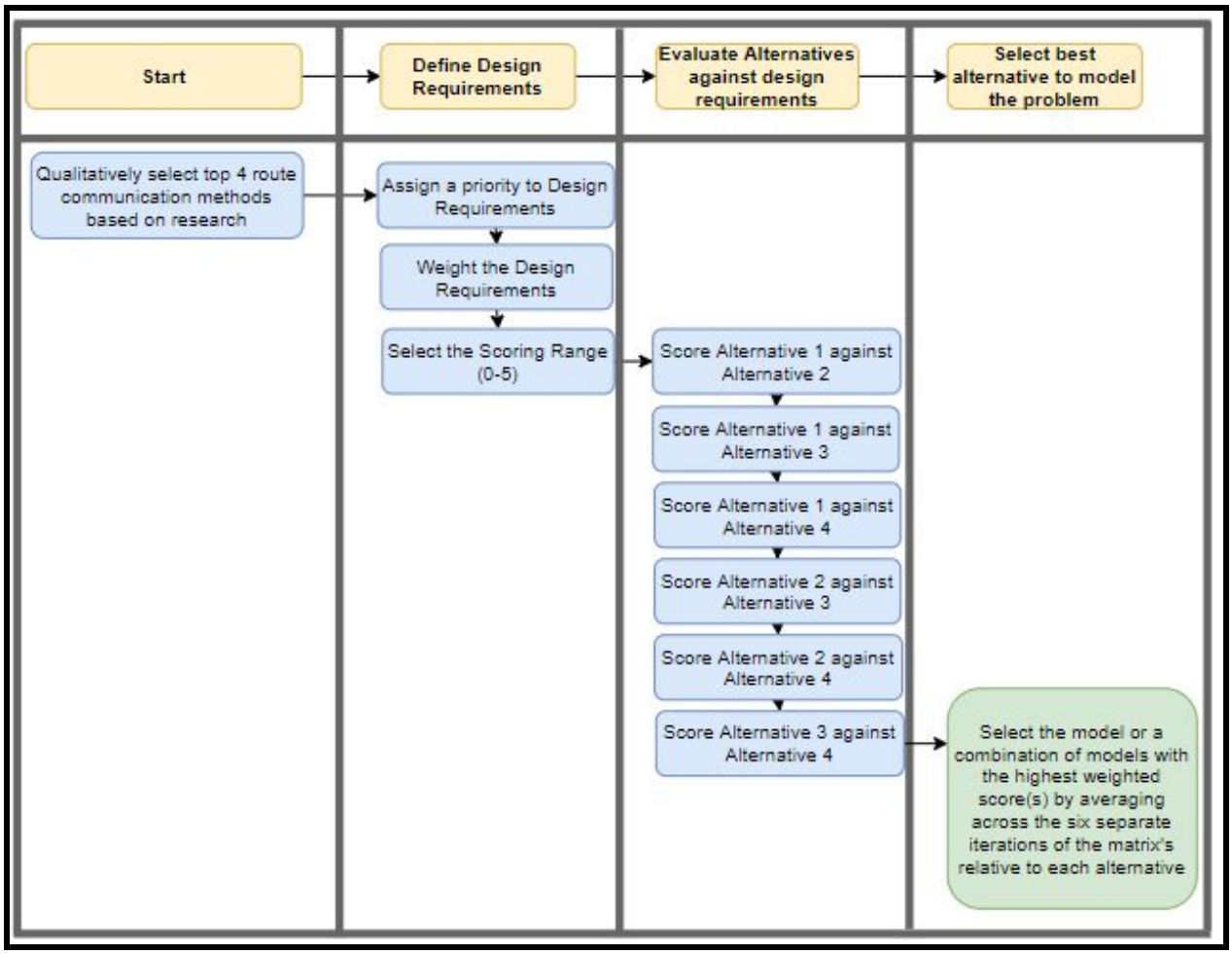


Figure 4.6 - The process for evaluation and selection of the best route communication alternative

During the design evaluation, four separate communication methods that were found to display characteristics that apply to this project's communication of new routes, as well as the original laminated paper maps the city currently uses, were evaluated using a matrix. This evaluation matrix is described in further detail below. The four models that were evaluated were: heads up displays, dash based GPS with audio, phone based GPS with audio, and the original paper maps.

To make the evaluation process as reliable as possible, a decision matrix was utilized to evaluate the alternatives. “Decision matrix techniques are used to define attributes, weigh them, and appropriately sum the weighted attributes to give a relative ranking among design alternatives. Note that, in practice, attributes are weighted as a numeric figure based on a prescribed ranking system for individual design alternatives. In some contexts, such as design optimization, attributes are also called design objectives, which are to be maximized or minimized, or constraint functions, which must be kept within limits. In general, attributes are also referred to as design criteria or decision criteria” (Chang, 2016).

This matrix not only weighted the design requirements relative to one another, it also assigned a score to each alternative based on a 6 point likert scale (0-6) in order to evaluate alternatives. This scale is as follows: (0=completely not satisfied, 1=mostly not satisfied, 2= somewhat not satisfied, 3=somewhat satisfied, 4=mostly satisfied, 5=completely satisfied). Each cell was multiplied by the weight of that column (design requirement), and this is why there are decimal values in each cell. The figures below, Figures 11-16, illustrate each iteration evaluating each alternative relative to another. Lastly, Figure 17 illustrates the averages that are then multiplied by the weighting of each design requirement weighting of those iterations into one summary graph where the final selection occurs.Calculations are shown in Appendix B.

ITERATION 1 - Comparing HUDs and Dash Based GPS

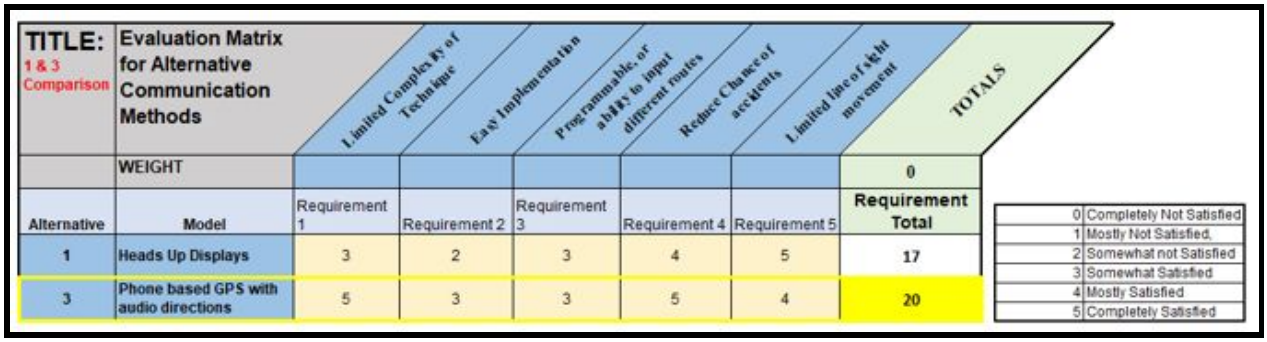
TITLE: 1 & 2 Comparison	Evaluation Matrix for Alternative Communication Methods	Limited Complexity of Technique	Easy Implementation	Programmable, ability to input different routes	Reduce Chance of accidents	Limited line of sight movement	TOTALS
	WEIGHT						0
Alternative	Model	Requirement 1	Requirement 2	Requirement 3	Requirement 4	Requirement 5	Requirement Total
1	Heads Up Displays	3	2	3	4	4	16
2	Dash based GPS with Audio Directions	5	3	3	5	5	21

0	Completely Not Satisfied
1	Mostly Not Satisfied,
2	Somewhat not Satisfied
3	Somewhat Satisfied
4	Mostly Satisfied
5	Completely Satisfied

Figure 4.7 - Matrix iteration 1 rating HUDs relative to dash based GPS with audio

Illustrates the iteration where the heads up display was scored relative to the dash based GPS with audio directions. The yellow highlight indicates that the dash based GPS with audio directions was the winner when scored against design requirements.

ITERATION 2 - Comparing HUDs and Phone Based GPS

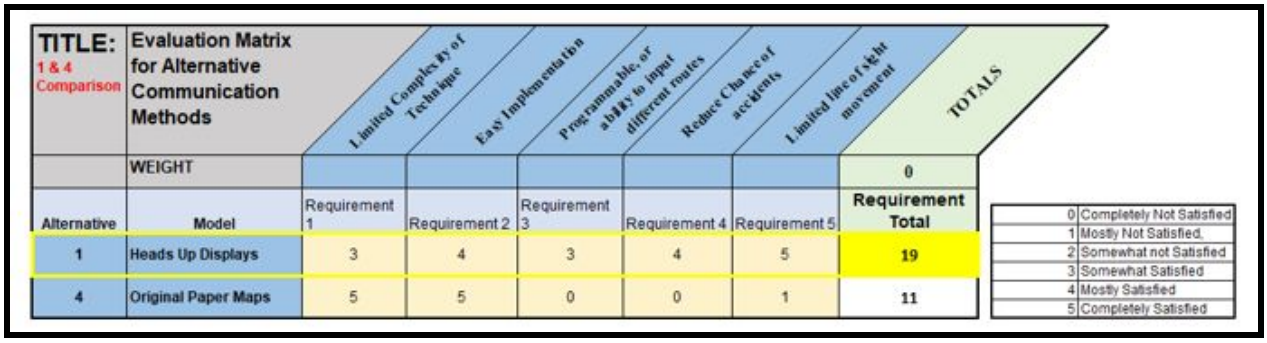


TITLE: 1 & 3 Comparison	Evaluation Matrix for Alternative Communication Methods	Limited Complexity of Technique	Easy Implementation	Programmable, of ability to input different routes	Reduce Chance of accidents	Limited line of sight movement	TOTALS
	WEIGHT						0
Alternative	Model	Requirement 1	Requirement 2	Requirement 3	Requirement 4	Requirement 5	Requirement Total
1	Heads Up Displays	3	2	3	4	5	17
3	Phone based GPS with audio directions	5	3	3	5	4	20

0	Completely Not Satisfied
1	Mostly Not Satisfied,
2	Somewhat not Satisfied
3	Somewhat Satisfied
4	Mostly Satisfied
5	Completely Satisfied

Figure 4.8 - Matrix iteration 2 rating HUDs relative to phone based GPS with audio

ITERATION 3 - Comparing HUDs and Original Paper Maps

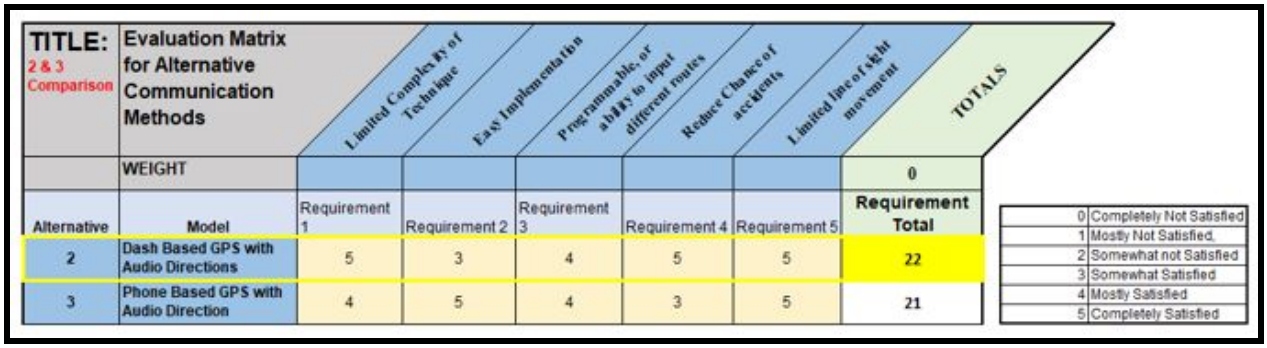


TITLE: 1 & 4 Comparison	Evaluation Matrix for Alternative Communication Methods	Limited Complexity of Technique	Easy Implementation	Programmable, of ability to input different routes	Reduce Chance of accidents	Limited line of sight movement	TOTALS
	WEIGHT						0
Alternative	Model	Requirement 1	Requirement 2	Requirement 3	Requirement 4	Requirement 5	Requirement Total
1	Heads Up Displays	3	4	3	4	5	19
4	Original Paper Maps	5	5	0	0	1	11

0	Completely Not Satisfied
1	Mostly Not Satisfied,
2	Somewhat not Satisfied
3	Somewhat Satisfied
4	Mostly Satisfied
5	Completely Satisfied

Figure 4.9 - Matrix iteration 3 rating HUDs relative to original paper maps

ITERATION 4 - Comparing Dash Based GPS and Phone Based GPS



TITLE: 2 & 3 Comparison	Evaluation Matrix for Alternative Communication Methods	Limited Complexity of Technique	Easy Implementation	Programmable, of ability to input different routes	Reduce Chance of accidents	Limited line of sight movement	TOTALS
	WEIGHT						0
Alternative	Model	Requirement 1	Requirement 2	Requirement 3	Requirement 4	Requirement 5	Requirement Total
2	Dash Based GPS with Audio Directions	5	3	4	5	5	22
3	Phone Based GPS with Audio Direction	4	5	4	3	5	21

0	Completely Not Satisfied
1	Mostly Not Satisfied,
2	Somewhat not Satisfied
3	Somewhat Satisfied
4	Mostly Satisfied
5	Completely Satisfied

Figure 4.10 - Matrix iteration 4 rating dash based GPS relative to phone based GPS with audio

ITERATION 5 - Comparing Dash Based GPS and Original Paper Maps

TITLE: 2 & 4 Comparison	Evaluation Matrix for Alternative Communication Methods	Limited Complexity of Technique	Easy Implementation	Programmable, or ability to input different routes	Reduce Chance of accidents	Limited line of sight movement	TOTALS
	WEIGHT						0
Alternative	Model	Requirement 1	Requirement 2	Requirement 3	Requirement 4	Requirement 5	Requirement Total
2	Dash based GPS with Audio Directions	4	4	4	5	5	22
4	Original Paper Maps	5	5	0	0	1	11

0	Completely Not Satisfied
1	Mostly Not Satisfied,
2	Somewhat not Satisfied
3	Somewhat Satisfied
4	Mostly Satisfied
5	Completely Satisfied

Figure 4.11 - Matrix iteration 5 rating dash based GPS relative to original paper maps

ITERATION 6 - Comparing Phone Based GPS and Original Paper Maps

TITLE: 3 & 4 Comparison	Evaluation Matrix for Alternative Communication Methods	Limited Complexity of Technique	Easy Implementation	Programmable, or ability to input different routes	Reduce Chance of accidents	Limited line of sight movement	TOTALS
	WEIGHT						0
Alternative	Model	Requirement 1	Requirement 2	Requirement 3	Requirement 4	Requirement 5	Requirement Total
3	Phone based GPS with audio directions	5	4	4	3	2	18
4	Original Paper Maps	5	5	0	0	1	11

0	Completely Not Satisfied
1	Mostly Not Satisfied,
2	Somewhat not Satisfied
3	Somewhat Satisfied
4	Mostly Satisfied
5	Completely Satisfied

Figure 4.12 - Matrix iteration 6 rating phone based GPS relative to original paper maps

ITERATION 7 - Summary Table of Average Scores from each iteration with weighting

TITLE: Summary Graph	Evaluation Matrix for Alternative Communication Methods	Limited Complexity of Technique	Easy Implementation	Programmable, or ability to input different routes	Reduce Chance of accidents	Limited line of sight movement	TOTALS
	WEIGHT	0.1	0.1	0.2	0.3	0.3	1
Alternative	Model	Requirement 1	Requirement 2	Requirement 3	Requirement 4	Requirement 5	Requirement Total
1	Heads Up Displays	0.3	0.3	0.6	1.2	1.4	3.8
2	Dash based GPS with Audio Directions	0.5	0.3	0.7	1.5	1.5	4.5
3	Phone based GPS with audio directions	0.5	0.4	0.7	1.1	1.1	3.8
4	Original Paper Maps	0.5	0.5	0.0	0.0	0.3	1.3

0	Completely Not Satisfied
1	Mostly Not Satisfied,
2	Somewhat not Satisfied
3	Somewhat Satisfied
4	Mostly Satisfied
5	Completely Satisfied

Figure 4.13 - Final Route Communication Summary Matrix iteration with averages

4.4.4 Selection of Route Communication Method

After evaluating the alternatives using an engineering tool called a design decision matrix, the MSU senior design team selected the best alternative amongst the four route communication methods, including the original paper maps, that were carried forward from the previous sprint after the route communications research was conducted. The selection process utilized the evaluation results from the decision matrix.

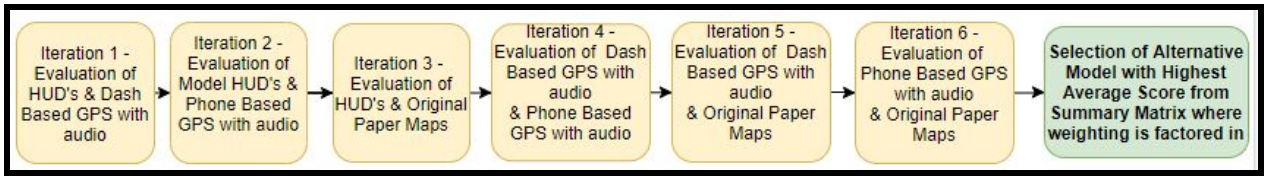


Figure 4.14 - Process of how the evaluation and selection are connected for route communication

The highest ranked alternative was the dash based GPS with audio directions, as seen in Figure 17. Heads up displays and phone based GPS with audio directions were both second. In research, phone based GPS with audio can utilize a device to become a heads up display, that may be why the scores came out together. Calculations are shown in Appendix B.

4.5 Recommendations

The team recommends the use of GPS units with audio directions for the route communication. Figure 4.14 illustrates the final evaluation matrix. In this matrix, all four methods of route communication are compared. Paper Maps scored a 1.3, Phone GPS and HUDs scored 3.8, and GPS with Audio scored the highest at 4.5. When compared to the paper maps that are currently in use, the GPS units have several major advantages. They limit line of sight movement, have the ability to program different routes, and have both visual and audio direction for the drivers. Eight GPS units will need to be purchased so each driver has access to one.

GPS units with audio directions can be purchased online. On Amazon, a Garmin GPS system with spoken turn by turn directions costs $99.99 brand new. Refurbished units are priced in the $60 to $80 range. These units come with the suction cup mounting and 12 volt power cable. Other mounting options can be purchased online.

4.6 Implementation Plan

An implementation plan is a list of steps that need to be completed to achieve a goal. For each step there is a section to record the start date, finish date, and status of the step. The team is providing the City of Bozeman with optimized routes in digital form. The team has recommended using GPS units with audio directions as the routing method to replace the paper maps. If the city decides to use GPS units, the implementation plan below (Table 4.2) defines the steps that the City of Bozeman should be taking to smoothly transition from using the original paper maps to GPS units with audio directions.

Table 4.2 - The Implementation Plan for Modified Route Communication Method

Activity		Start Date	Finish Date	Responsibility
1	**Purchase 8 GPS units**			
2	**Confirm software is up to date**			
3	**Load Optimized Snow Plow Routes onto devices**			
4	**Confirm all trucks have working 12 volt power outlet**			
5	**Fix any faulty 12 volt power outlets**			
6	**Schedule Driver GPS training session**			
7	**Plow Driver GPS training session**			
8	**Provide each truck with mounting solution**			
9	**Drivers mount/install GPS with suction cup to windshield**			

10	**Run power cable from GPS unit to 12 volt supply**			
11	**Confirm voice navigation is enabled**			
12	**Distribute Plow Route assignment to drivers**			
13	**Driver select assigned plow route in GPS unit**			
14	**Test run plow routes with GPS visual and audio communication**			

The final handoff occurs when the MSU team provides the City of Bozeman with the optimized snow plow routes. After the routes are handed off, the MSU team is not responsible for how the city decides to implement the routes. The implementation plan above has intentionally been left blank in the Start Date, Finish Date, and Status columns. The team suggests implementing the use of GPS units before the next winter. Test runs can be made during the summer months to confirm the drivers are comfortable with this method of route communication. There are 14 steps in the plan that lay out how to transition from using paper maps to GPS units. A total of eight GPS units must be purchased as each plow truck will need one. The optimized routes will need to be loaded onto the devices, so that the user can pick their assigned route and begin their shift. Most aftermarket GPS units use a 12 volt power supply so each truck must have a working power outlet. The team suggests a training session in which the drivers are given instructions on how to use the GPS units and navigate the menus. Once the plow drivers feel comfortable with the GPS units, they can be mounted in the vehicles and test runs can be made. Additional steps can be added to the implementation plan if needed.

5. Economic Analysis

5.1 Overview

Although economic considerations were not the focus of this project, it was agreed with the client that any proposed solutions should limit spending when possible. Additionally the solutions could have broader impacts outside of immediate costs or savings.

5.2 Operations Cost Analysis

The estimated cost of the proposed route communications method is $99 per vehicle, for a total of $792 to equip all 8 vehicles. This is a relative cost increase over the current paper communication method which is limited by printing and laminating in-house and generally requires reprints at most once per year to accommodate route changes. However, Costs due to accident repairs could be reduced by allowing drivers to keep their eyes on the road while receiving directions from the GPS units.The cost to train new operators is extremely high, as the city offers new operators with commercial drivers training in addition to the specific training required for snow plow operation. If the investment of $693 influences even one new hire to continue working for the city that translates to thousands of dollars in savings from training costs.

The economic impact due to optimized routes is harder to define. Even if priority routes are completed more quickly, drivers will still work a full shift, only now plowing residential streets. Thus the impact on operations costs such fuel should be negligible. However, optimized routes could lead to a reduction in need for overtime as more work would be completed during the standard shift. If optimized routes lead to priority streets being cleared before 7:00am, this could have wider economic impacts in terms of time saved by commuters, reduced accidents, and increased commerce due to safer roads.

6. Broader Engineering Design Considerations

6.1 Overview

Although many of the broader design considerations such as environmental impact and sustainability were not included in the project's design objectives, the design team considers them important factors that must be accounted for.

6.2 Broader Design Considerations for the Routing Model

Environmental & Sustainability

In terms of the Model:

If it proves effective, the route optimization model should increase the efficiency of travel for snow plows, leading to less drive time and less vehicle emissions overall. Additionally, if the optimized routes lead to less overlap of plowing priority roads during morning traffic, commuters will be able to reach their destination more efficiently and emissions will be reduced in that regard as well. There are no negative bi-products that can be foreseen as result of optimized snow plow routing, as it does not involve the use of any new materials or increase the use of gas, salt or sand. The lifecycle of the route optimization solution is only as long as the priority routes do not change. However, the model could have any changes to the priority streets incorporated in it, extending its useful life indefinitely. The routing model will only be implemented if it does prove superior in efficiency to the current solution and thus can only improve or remain the same in terms of sustainability.

In terms of the Recommended Route Communication Method:

The recommended route communication method of dash based GPS with audio directions will aid in the implementation of the newly optimized routes. In addition to aiding, the overall efficiency is expected to improve. With reduced times, there is opportunity for less emissions released from the snow plows because they are taking less time to finish the job., ultimately resulting in an improvement in regards to environmental considerations.

Globalization

In terms of the Model:

This solution has no immediate impact on people worldwide, however it could be implemented anywhere, with positive results, if the relevant street data was altered. If the model is successful it could delay the city's need to purchase additional plows thus reducing reliance on global infrastructure, if only slightly. This would have an equally slight negative impact on the global economy. In general the design team would anticipate different communities and cultures to view this solution in a positive way. However, it may be seen by some as a privileged endeavor, that determining routes using a commonsense approach is sufficient and that the time and resources devoted to developing this solution could have been better spent elsewhere.

In terms of the Recommended Route Communication Method:
Through the lens of globalization, the route communication method may not impact the worldwide market, as it was tailored to be specifically for Bozeman, MT. There is a large body of research with similar projects, and this is just another example across the globe of how operations research and mathematical models can optimize different routes. Globally, Bozeman, MT is doing it's part to ensure that it is reducing emissions whenever opportunity arises. The method of communication of the routes uses the global positioning of the snow plows to provide valid directions.

Social Factors
In terms of the Model:
If effective, this solution will certainly benefit society as a whole. Bozemen residents and visitors will spend less time driving slowly behind snow plows, and will be driving on safer roads more of the time. By reducing the time required to plow priority roads, there will be more time available to attend to residential streets, making driving safer and easier across the board. The solution directly addresses public safety by making snow plow operators work more efficiently, road safety will be higher over the course of the winter. This solution may not be equitable for snow plow operators if it leads to a reduction in work. However, the increases in efficiency are not anticipated to be so extreme as to reduce the need for snow plow operators, but simply allow them to get slightly more work done in the time they currently have. Employee satisfaction amongst snow plow operators is somewhat stratified, with some operators working for the city for decades, while many leave after just one season. The design team does not anticipate route optimization having a large impact on employee satisfaction, but if it has any it should be positive as employees will see their work being more effective in general.

In terms of the Recommended Route Communication Method:
There are several social factors that are affected by the newly designed route communication method. The number one social factor is public safety. If the drivers continue to utilize paper maps to learn a new route, there is opportunity for accidents because the driver's eyes may not be on the road enough. It is imperative that the new routes do not forego any of the safety to the public.

Cultural Factors
In terms of the Model:
The proposed routing solution fits well with the organizational culture of the Bozeman Streets Department. Under Superintendent John Van Delinder the department encourages an environment of continuous improvement and many operators take pride in the value their work provides for the city and would be happy to see an increase in efficiency. However, there is some potential for resistance to the proposed solution because some operators may be reluctant to change from routes they have plowed for years and be skeptical of the value of the new routes. The current culture was not considered an obstacle during the routing model design process. Design team members did consider that new routes, if sufficiently different, might prove challenging to "sell", however the solution produced does not suggest major changes and is not anticipated to be challenging for the current culture to accept.

In terms of the Recommended Route Communication Method:
Cultural factors that were considered with regards to the route communication method were primarily from the fact that new optimized routes result in having to change things about the current system. Several drivers have been doing the same route for years, and changing the culture of route communication may lead to push back from them. If the new communication method is effective, it can help mitigate any of that hesitation to switch over to the new optimized routes.

Ethical

In terms of the Model:
Implementation of the proposed routing solution does not pose any major ethical questions. If successful, the improved routes are anticipated to improve the welfare of everyone who drives in bozeman, which aligns well with the first fundamental principle of the Institute of Industrial & Systems Engineers (IISE) which is, "Using their knowledge and skill for the enhancement of human welfare." Additionally, as the design team has worked hard to provide a useful solution that will demonstrate the value of the industrial engineering profession, it also fits with IISE's third fundamental principle which is, "Striving to increase the competence and prestige of the engineering profession."

In terms of the Recommended Route Communication Method:
The team took ethics into account when designing a new route communication method by ensuring that the product (GPS unit) is not made in unethical manners, as well as the research into the prospective models is ethical. The implementation must also be ethical in the eyes of the local city government.

7. Project Management

7.1 Overview

For the management of the project, the team utilized the agile project management method. The project was divided into seven separate sprints, each having its own scrum master. The scrum master is the leader of the group and is responsible for; “Ensuring that goals, scope, and product domain are understood by everyone on the Scrum Team as well as possible, finding techniques for effective Product Backlog management helping the Scrum Team understand the need for clear and concise Product Backlog items, understanding product planning in an empirical environment, ensuring the Product Owner knows how to arrange the Product Backlog to maximize value, understanding and practicing agility, and facilitating Scrum events as requested or needed” (Scrum, 2020). The sprints were treated as smaller projects with clearly defined deliverables to be completed by the end of the sprint. Each sprint was assigned to a different team member in order to delegate the responsibility. At the end of each sprint, the team would provide the deliverables to the project client, Superintendent Van Delinder, who would provide feedback. The deliverables from each of the seven sprints are contained within the final report.

7.2 Stakeholder Communication

Stakeholder Meetings

Stakeholder meetings were held at the end of every sprint. The meetings are “a strategic way to derive usability objectives from business objectives, and to gain commitment to usability”(Wilson, 2012). Before each meeting, an agenda was created detailing what was going to be covered during the meeting.The team met with Superintendent Van Delinder to discuss the progress of the project and gain feedback on the deliverables that were provided to the client. By meeting with the client often, the team and the client had a shared vision on how the project would progress and the same goals. The benefits of stakeholder meetings are that, “all factors that relate to use of the system are identified before design work starts and together all the people relevant to the development, to create a common vision. (Wilson, 2012).

Stakeholder Presentation

For this project, the stakeholders were Street Superintendent Van Delinder, the plow drivers, and the City of Bozeman. The final step in this project was the Stakeholder Presentation, where the solutions and methods to reach the solutions presented to the stakeholders. The presentation occured on May 7th, 2020 over video call. The MSU team presented their optimized snow plow route solution to the client, as well as the recommendations for implementation of both the optimized routes and the route communication method. The end of this meeting was the official hand off of the project from the MSU team to the City of Bozeman.

7.3 Agile Project Management

Agile Project Management is the management method that the team used during this project. “Agile project management is deeply rooted in these principles but slightly modified to make sense in the project management, rather than software development, environment. This can be seen in some of the qualities of

the agile project management approach. For example, agile project management emphasizes two important concepts. The first is that risk is minimized by focusing on short iterations of clearly defined deliverables. The second is that direct communication with partners in the development process is emphasized in lieu of creating copious project documentation. The reasons these two concepts are emphasized is simple: both help a project team adapt quickly to the unpredictable and rapidly changing requirements most development projects are carried out in" (Cervone, 2010). Agile project management utilises the Scrum model, laid out as follows, "The Scrum process has five major activities: the kickoff, the sprint planning meeting, the sprint, the daily Scrum, and the sprint review meeting" (Cervone, 2010).

Sprint Planning Meetings

Sprint planning meetings are a critical part of Agile Project Management. At the beginning of each sprint, the team meets to determine the deliverables and product backlog for the sprint. There are two steps that occur at a sprint planning meeting, "In the first part of the meeting, two major activities occur. First, the group defines the product backlog, which is basically a list of the project requirements. After this, the group determines the sprint goal, which is the formal outcome(s) from this particular sprint" (Cervone, 2010). The project was split into seven different sprints, each with their own deliverables. At the sprint planning meetings, the team split the deliverables into tasks that needed to be completed. A scrum board and earned value reporting were used to keep track of the tasks each sprint, and were created at each sprint planning meeting.

The Sprint

The project was broken down into seven smaller sections referred to as sprints. These sprints lasted two weeks and were treated as small projects. As stated earlier, each sprint is led by the scrum master. The sprint planning meetings were used to define the goals of each sprint, and the deliverables that would be presented to the client. Examples of these deliverables are the Sprint A3 Report, Sprint Final Report, and the Route Communication Report. A Scrum Board was used to track sprint progress and delegate tasks to each member. Once a sprint has started, "the project requirements cannot be changed"(Cervone, 2010). If tasks were not completed by the end of the sprint, they were pushed to the next sprint. The A3 report and Earned Value Report were completed at the end of each sprint to track project progress.

The Scrum Board

Physical:

For the start of the project, the Scrum board was a physical double-sided white board on wheels. The board was used to monitor project progress and delegate deliverable tasks between the team and group members. The front side of the board was broken down into five columns: Party Responsible, To Do, In Progress, In Review, and Completed. The team and each member had their own column. Sticky notes with a short description of the task were created for each task that needed to be completed. The sticky notes started the sprint in the "To Do" column and were moved to "In Progress" when started, and "completed" when the task was finished. The back side of the scrum board housed the overall project burn up chart and the schedule of scrum master changes. The sprint performance chart on the back was updated at the end of each sprint.

Virtual:
For the end of the project, due to the face to face educational model being updated to distance learning, the scrum board was made virtual on a platform called iObeya. The team was able to recreate the scrum board on iObeya with an identical layout to the physical board that was used earlier in the semester. The same method of task distribution was used. With iObeya, the "sticky notes" had a description section which was a feature that is not possible with a physical sticky note. The team used these description sections to give more guidance on the tasks that needed to be completed. Much like the physical board, the tasks were moved when they were in progress and completed. The burn up chart from the backside of the physical scrum board was a second window in iObeya.

Scrum Meetings
At the beginning of the semester, the team held stand up scrum meetings every Monday, Wednesday, and Friday. Scrum meetings are used to determine what work has been completed, what is being worked on, and what challenges are being faced. In practice, "This meeting, typically lasting no more than 15 minutes, is held every day between the Scrum master (who chairs the meeting) and the Scrum team. In this meeting, every team member briefly answers three questions: (1) What did you do since the last Scrum? (2) What are you doing until the next Scrum? (3) What is stopping you getting on with your work?" (Cervone, 2010). The team's meetings lasted between 10 and 30 minutes on average and were run by the sprint scrum master.

Strategies
For this project, the team used three different strategies; Incremental Strategy, Iterative Strategy, and Adaptive Strategy. An Incremental Strategy is, "A strategy that consists of dependent, sequential phases that are executed with no feedback loops. Each phase of the project releases a partial solution" (Fernandez, D., 2008; Fernandez, J., 2008). Each sprint is considered an incremental strategy. They are sequential phases and need to be completed before the next one can begin. An Iterative strategy, "consists of a number of repeated phases that include a feedback loop after a group of phases is completed. The last phase of a group may include a partial solution if the customer desires. The iterative strategy is a learn-by-doing strategy that uses intermediate solutions as a pathway to discover the details of the complete solution" (Fernandez, D., 2008; Fernandez, J., 2008). The sprints are also
When using Adaptive strategy, "each iteration's feedback adjusts the next iteration so that a solution will be converged upon. To remove the uncertainty, the solution is arrived at via a continuous change process from iteration to iteration. The success of the Adaptive strategy is therefore highly dependent on the ability to accommodate frequent change and adjust accordingly. Therefore, planning is done primarily in a just-in-time manner" (Fernandez, D., 2008; Fernandez, J., 2008).

7.4 Earned Value Reporting

The design team utilized earned value reporting to document and communicate progress during the project. Earned value reporting combines information on schedule, cost and work performance to quantify project status(Czarnigowska, Jaskowski, & Biruk, 2011). It involves creating a budget of planned time, often quantified by a dollar cost of that time, for a specific work period and comparing that to the amount of time actually spent during that period (McConnell, 1985). Earned value reporting is a widely accepted

technique and according to Brandon, “Earned value is a powerful approach for evaluating true project performance. It also provides a quantitative basis for estimating actual completion time and actual cost at completion” (Brandon, 1998).

7.5 A3 Reporting

As a means to communicate the status of the project to both the client, Superintendent Van Delinder, and the advisors to the MSU senior design team, a status A3 was created as a deliverable for all sprints except sprint 1 and 7 because they were the initial sprint and final sprint. In sprint 1, the team was primarily concerned with the statement of work, and sprint 7 was primarily project handoff and final deliverables. “The A3 report is a tool that Toyota Motor Corporation uses to propose solutions to problems, give status reports on ongoing projects, and report results of information gathering activity” (Sobek & Jimmerson, 2004). A3 thinking is also an approach, and not just a document giving the status of a project. [A] metaroutine such as the A3 process can be very effective for organizational members to collectively validate existing knowledge through shared understanding, identify and deal with the problems at their sources, and create new knowledge to address them for a sustainable change” (GHOSH & Sobek, 2002). “A3 Thinking is fundamental to Toyota’s benchmark management philosophy and to their lean production system. It is used to solve problems, gain agreement, mentor team members, and lead organizational improvements. A structured problem-solving approach, A3 Thinking builds improvement opportunities through experience” (Anderson, Morgan, & Williams, 2011). “Conceptually, A3 is a series of boxes that tell the story of what is to be communicated from upper left hand corner to lower right hand corner. Various formats for A3 reports have been developed to communicate different types of information in many different industries who have adopted the practice. However, Toyota is quick to point out that the beauty of A3 Thinking is that it is not the format that matters, but the process that guides the user and the audience clearly and visually through the PDCA thought process” (Loyd, Harris, & Blanchard, 2010). The team found this method to be very beneficial in communicating any problems that arose, as well as a way to think through each sprint. The PDCA thought process proved to be an approach worth using on programming projects like this.

7.6 Overall Project Performance

Project Performance & Earned Value Metrics

Figure 7.1 and Table 7.1 illustrate the performance metrics generated using earned value over the course of the entire project duration.

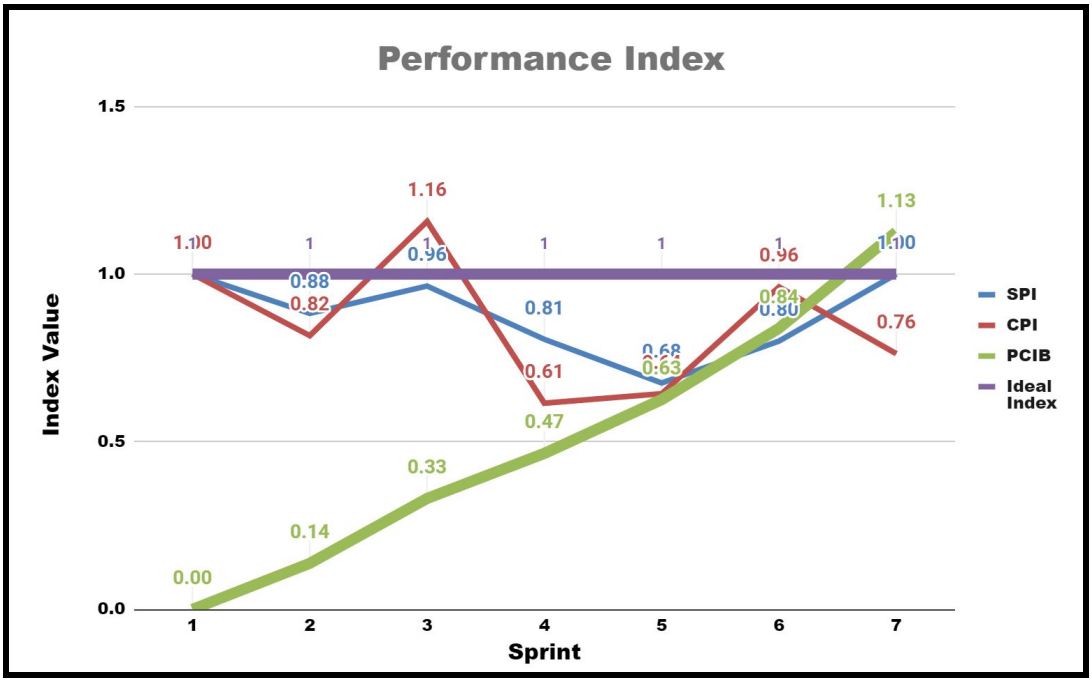


Figure 7.1 - Performance Index Chart for Project Tracking

Table 7.1 - Overall Project Metrics & Final Sprint Summary Table

Metric	Sprint 7	Overall Project
Schedule Performance Index (SPI) **(On Schedule = 1)**	1.00	1.05
Cost Performance Index (CPI) **(On Cost = 1)**	0.76	0.80
Percent Complete in Terms of Budget (PCIB) **(100% of budget = 1.00)**	1.13	1.13

- **SPI -** This Schedule Performance Index **(SPI) = 1.05** indicates that the project was 5% ahead of schedule at the end of the project.
- **CPI -** This Cost Performance Index **(CPI) = 0.80** indicates that the project is slightly over budget.
- **PCIB -** Percent complete (PCIB) indicates the project is **113%** complete (in terms of the budget).

This project was (5%) ahead of schedule, (20%) over budget, and about 113% complete in terms of the budget.

8. Conclusions

The design team was able to produce optimized routes for segmented sections of the city using the optimization model laid out in section three of this report. Moving forward it is recommended that the City of Bozeman consider the proposed route changes seen in figure 3.15. In order to confirm that the model is valid beyond the mathematical confirmation it would be necessary to compare the actual drive time of the new routes to the existing. Furthermore, it is recommended that the city work with an engineering consultant or future capstone team in order to run the full final model and continue to improve its accuracy.

The team recommends the use of GPS units with audio directions for the route communication. Identified in Figure 4.14 as the superior option. We anticipate that this solution will lead to safer snow plow operation and make the job easier to learn for new employees and those who do not reside in Bozeman. The relatively low cost of this solution combined with the potential savings from reduced accidents and employee turnover makes this solution well worth implementation

In conclusion the design team is proud of the quantity and quality of work that was put into developing these solutions. Although the route model was not completed to the degree that the design team originally desired, they are confident that it can provide great value to the City of Bozeman as they continue to improve their snow removal methods.

References

"1981–2010 Normals". *National Oceanic and Atmospheric Administration*. Retrieved from https://www.ncdc.noaa.gov/cdo-web/datatools/normals

AMPL [Computer software]. (2020).

Anderson, J. S., Morgan, J. N., & Williams, S. K. (2011). Using Toyota's A3 Thinking for Analyzing MBA Business Cases. *Decision Sciences Journal of Innovative Education*, *9*(2), 275-285.

Andreescu, M., & Frost, D. B. (1998, February 27). Weather and traffic accidents in Montreal, Canada. *Climate Research*, *9*, 225-230. doi:https://doi.org/10.3354/cr009225

Andrew.currin.ctr@dot.gov. (2020, March 6). U Drive. U Text. U Pay. Retrieved from https://www.nhtsa.gov/risky-driving/distracted-driving

Aziz, E. E. (2015). Project closing the small process group with big impact. In *Project Management Institute*.

Bertsimas, D., Delarue, A., & Martin, S. (2019). Optimizing schools' start time and bus routes. *Proceedings of the National Academy of Sciences*, *116*(13), 5943-5948.

Bozeman Population. (2019-10-29). *World Population Review.* Retrieved from http://worldpopulationreview.com/us-cities/bozeman-population/

Brandon Jr, D. M. (1998). Implementing earned value easily and effectively. *Project Management Journal*, *29*(2), 11-18.

Cervone, H. F. (2011). Understanding agile project management methods using Scrum. *OCLC Systems & Services: International digital library perspectives*.

Chang, K. H. (2016). *e-Design: computer-aided engineering design*. Academic Press

Chen, H. (2015). *Lignocellulose biorefinery engineering: principles and applications* (No. 74). Woodhead Publishing.

Christensen, D. S. (1998). The costs and benefits of the earned value management process. *Journal of Parametrics*, *18*(2), 1-16.

Christensson, P. (2006). GPS Definition. Retrieved 2020, Apr 19, from https://techterms.com

Christofides, N., Campos, V., Corberan, A., & Mota, E. (1986). AN ALGORITHM FOR THE RURAL POSTMAN PROBLEM ON A DIRECTED GRAPH . In Mathematical Programming Study 26 (1986) 155-166 . Retrieved from https://link.springer.com/content/pdf/10.1007%2FBFb0121091.pdf

CPLEX [Computer software]. (2020).

Czarnigowska, A., Jaskowski, P., & Biruk, S. (2011). Project performance reporting and prediction: extensions of Earned value management. *International journal of business and management studies*, *3*(1), 11-20.

Dolan, E. 2001. The NEOS Server 4.0 Administrative Guide. Technical Memorandum ANL/MCS-TM-250, Mathematics and Computer Science Division, Argonne National Laboratory.

Dowds, J., Sullivan, J., Novak, D. C., & Scott, D. (2016). *Identifying best practices for snowplow route optimization* (No. CR 14-07). Minnesota. Dept. of Transportation. Research Services & Library. Du, T., Wang, F. K., & Lu, P.-Y. (2006, June 5). A real-time vehicle-dispatching system for consolidating milk runs. Retrieved from https://www.sciencedirect.com/science/article/pii/S1366554506000287

Enrique Benavent, David Soler, (1999) The Directed Rural Postman Problem with Turn Penalties. Transportation Science 33(4):408-418. https://doi.org/10.1287/trsc.33.4.408

Fernandez, D. J., & Fernandez, J. D. (2008). Agile project management—agilism versus traditional approaches. *Journal of Computer Information Systems*, *49*(2), 10-17.

Fourer, R., Gay, D. M., Kernighan, B. W. (2003). *AMPL: A Modeling Language for Mathematical Programming*. Retrieved from https://ampl.com/resources/the-ampl-book/chapter-downloads

GHOSH, M., & Sobek, D. (2002). Effective metaroutines for organizational problem solving. *Mechanical and Industrial Engineering Department, Bozeman*.

GPS Accuracy. (2017, December 5). Retrieved March 13, 2020, from https://www.gps.gov/systems/gps/performance/accuracy/

GPS Tracking Solutions, (2017, June 18). In *The Advantages of GPS Tracking System*. Retrieved from https://www.gpswox.com/en/blog/useful-information/the-advantages-of-gps-tracking-system

H. A. Eiselt, Michel Gendreau, Gilbert Laporte, (1995) Arc Routing Problems, Part II: The Rural Postman Problem. Operations Research 43(3):399-414. https://doi.org/10.1287/opre.43.3.399

Harris, Sean. Personal interview. 9 Mar. 2020.

Hildenbrand, J. (2018, August 24). How does GPS work on my phone? Retrieved April 3, 2020, from https://www.androidcentral.com/how-does-gps-work-my-phone

Hoffman, K. L., Padberg, M., & Rinaldi, G. (2013). Traveling salesman problem. *Encyclopedia of operations research and management science*, *1*, 1573-1578.

How Fast is Bozeman Growing? (2019, August 2). In *Bozeman Real Estate Group*. Retrieved from https://bozemanrealestate.group/resources/blog/how-fast-is-bozeman-growing

HUDWAY Drive. (n.d.). Retrieved April 2, 2020, from https://store.hudway.co/drive?utm_source=site&utm_medium=cp&utm_campaign=hwdrive&_ga=2.158465587.803889942.1585898359-1784062602.1585898359

ITO, H., UNO, H., Atsumi, B., & Akamatsu, M. (2001). Visual Distraction while Driving. *Iatss Research*, *25*(2), 21.

Kendall, L. (2018, March 29). Bozeman area again ranked fastest-growing of its size in the nation. *Bozeman Daily Chronicle*. Retrieved from https://www.bozemandailychronicle.com/news/bozeman-area-again-ranked-fastest-growing-of-its-size-in/article_881f7994-a3d7-5f68-b85c-5875d2c0749a.html

Khalid, H. M., & Helander, M. G. (2004). A framework for affective customer needs in product design. *Theoretical Issues in Ergonomics Science*, *5*(1), 27-42.

Kinable, J., van Hoeve, W. J., & Smith, S. F. (2016, May). Optimization models for a real-world snow plow routing problem. In *International Conference on AI and OR Techniques in Constraint Programming for Combinatorial Optimization Problems* (pp. 229-245). Springer, Cham.

Kodosky, J. (2010). Q&A: Why System-Level Design Is the Future of Engineering. In *System Level Design Q&A*.

Korteweg, P., & Volgenant, T. (2011, January 13). On the Hierarchical Chinese Postman Problem with linear ordered classes. Retrieved from https://www.sciencedirect.com/science/article/pii/S037722170400428X

Liu, G., Ge, Y., Qiu, T. Z., & Soleymani, H. R. (2014). Optimization of snow plowing cost and time in an urban environment: A case study for the City of Edmonton. *Canadian Journal of Civil Engineering*, *41*(7), 667-675.

Loyd, N., Harris, G. A., & Blanchard, L. (2010). Integration of A3 thinking as an academic communication standard. In *IIE Annual Conference. Proceedings* (p. 1). Institute of Industrial and Systems Engineers (IISE).

McConnell, D. R. (1985). Earned value technique for performance measurement. *Journal of Management in Engineering*, *1*(2), 79-94.

Nichols, Megan Ray. (2019, September 4). How cities can optimize snow removal to boost road safety and save dollars. *Roads & Bridges.* Retrieved from https://www.roadsbridges.com/how-cities-can-optimize-snow-removal-boost-road-safety-and-save-dollars

Nossack, J., Golden, B., Pesch, E., & Zhang, R. (2016, September 10). The windy rural postman problem with a time-dependent zigzag option. Retrieved from

Ontiveros, F. (2008). Retrieved from https://apps.dtic.mil/docs/citations/ADA487563

Pfeifer, M. (2009). *Materials enabled designs: The materials engineering perspective to product design and manufacturing*. Butterworth-Heinemann.

Salazar-Aguilar, M. A., Langevin, A., & Laporte, G. (2012). Synchronized arc routing for snow plowing operations. *Computers & Operations Research*, *39*(7), 1432-1440.

Satellite Navigation - Global Positioning System (GPS). (2014, November 13). Retrieved from https://www.faa.gov/about/office_org/headquarters_offices/ato/service_units/techops/navservices/gnss/gps/

Scrum, (2020). What is a Scrum Master? Retrieved from https://www.scrum.org/resources/what-is-a-scrum-master

Snow Plowing 101 (n.d.). *Bozeman MT*. Retrieved from https://www.bozeman.net/government/streets/snow-plowing

Sobek, Durward. "Engineering Design Process." Montana State University, Feb. 2020, Bozeman. Lecture.

Sobek II, D. K., & Jimmerson, C. (2004). A3 reports: tool for process improvement. In *IIE Annual Conference. Proceedings* (p. 1). Institute of Industrial and Systems Engineers (IISE).

This Futuristic Smartphone Display Lets You Navigate Safely & with Better Visibility. (n.d.). Retrieved April 2, 2020, from https://shop.popsci.com/sales/hudway-glass-heads-up-display?gclid=CjwKCAjwguzzBRBiEiwAgU0FT9w8DDp5nIMDNAwsgFqGE8rcG_MGb8JxB1nBERWPzXOR2CYzawj7TxoCl48QAvD_BwE

Wallman, Carl-Gustaf, Wretling, Peter & Gudrun Öberg. (1997). *Effects of Winter Road Maintenance.* Retrieved from http://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.624.5238&rep=rep1&type=pdf

Wickens, C. D., & Ververs , P. M. (1998). Retrieved from https://www.faa.gov/data_research/research/med_humanfacs/oamtechreports/1990s/media/AM98-28.pdf

Wilson, C. (n.d.). Stakeholder meeting: UsabilityNet. Retrieved from https://www.usabilitest.com/usabilitynet/tools-stakeholder

Zabrodin, R. (2015). The Route of the Postman. Retrieved from https://www-m9.ma.tum.de/graph-algorithms/directed-chinese-postman/index_en.html

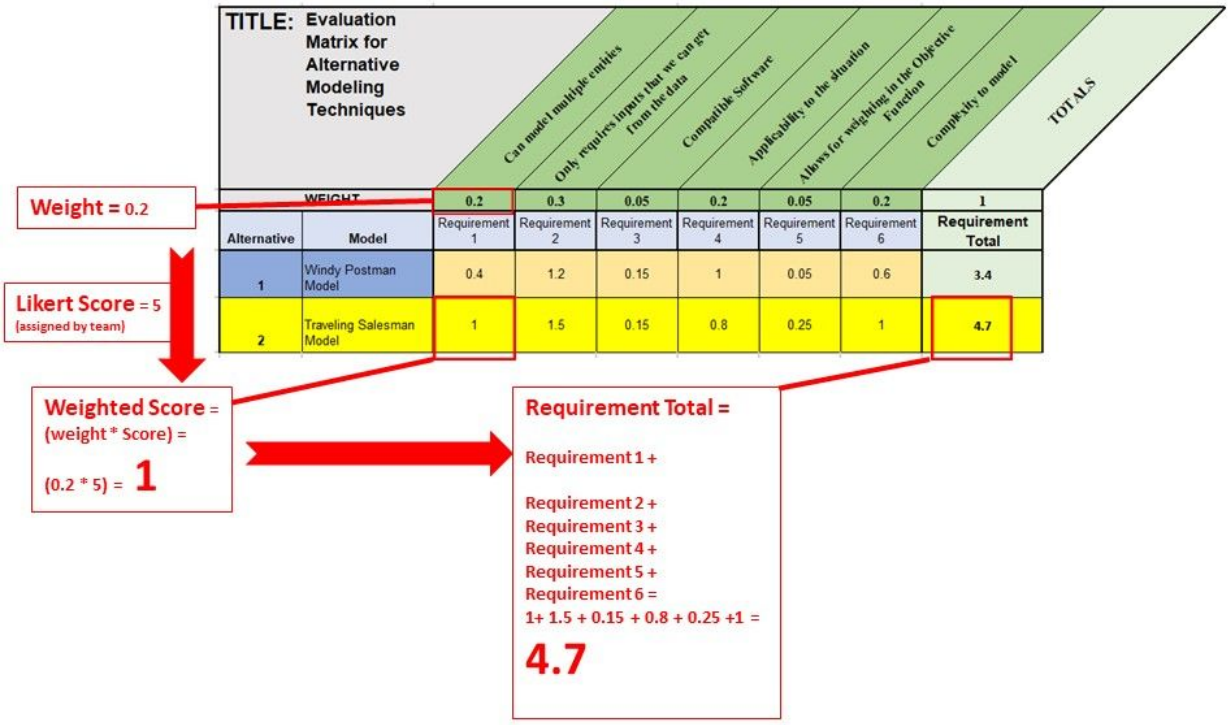
Zeng, Y., & Gu, P. (1999). A science-based approach to product design theory Part II: Formulation of design requirements and products. *Robotics and Computer-Integrated Manufacturing*, *15*(4), 341-352.

Appendices

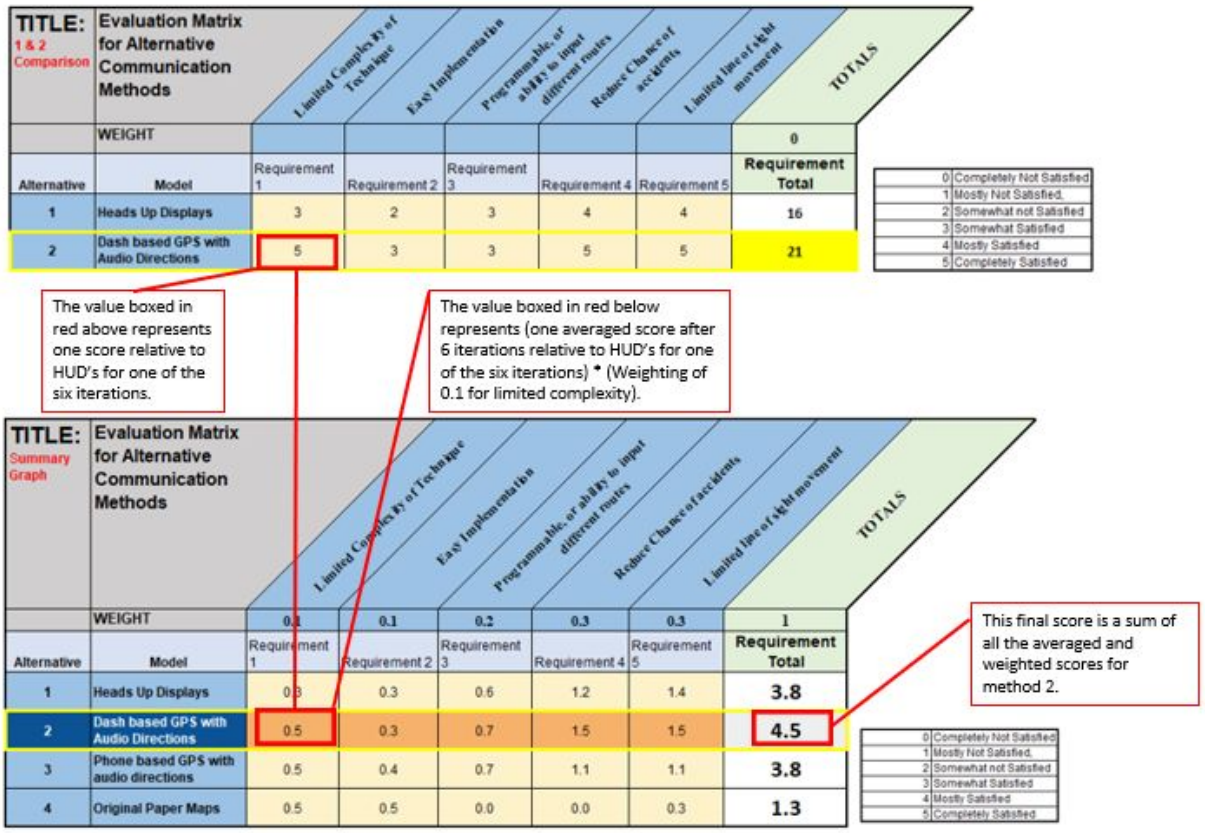
Appendix A - Engineering Concepts used from MSU Curriculum

MSU Class	Engineering Concept	Application to Project
EIND 464 - Principles of Operations Research II	Operations Research Concepts	The team utilized the methodology of operations research to create a network that represented Bozeman's streets with an arc, and turns/intersections as a node.
EIND 442 - Facility and Material Handling Systems Design	Engineering Design Matrix	The team used the engineering design matrix for two separate designs: the optimization model and the route communication method. This was a way to narrow down several alternatives into the best one with regards to the design requirements developed.
EIND 413 - Ergonomics & Human Factors Engineering	Human Factors & Ergonomics	The team utilized the concept of line of sight to design a safe way to communicate the route communication methods. This methodology led the team to a dash based GPS with audio directions system.
EIND 410 - Interaction Design	Human Factors & Ergonomics	The team utilized the concept of interaction design because the snow plow drivers are the end user of the product. This means they needed a way to learn new routes as easily as possible.
EIND 371 - Computer Integrated Manufacturing	Computer Programming	The team utilized computer programming to sort through and organize data, model the network, and gather outputs that make optimized routes for the snow plows.
EIND 300 - Engineering Management & Ethics	Engineering Design & Ethics Concepts	For broader design considerations, the team integrated an ethical thought process regarding several different areas: Environmental, Globalization, Social Factors, and cultural factors.

Appendix B - Example Calculations for Model Alternative Decision Matrix



Appendix C - Example Calculations for Route Communication Alternative Decision Matrix



TITLE: 1 & 2 Comparison	Evaluation Matrix for Alternative Communication Methods	Limited Complexity of Technique	Easy Implementation	Programmable, or ability to input different routes	Reduce Chance of accidents	Limited line of sight movement	TOTALS
	WEIGHT						0
Alternative	Model	Requirement 1	Requirement 2	Requirement 3	Requirement 4	Requirement 5	Requirement Total
1	Heads Up Displays	3	2	3	4	4	16
2	Dash based GPS with Audio Directions	5	3	3	5	5	21

0	Completely Not Satisfied
1	Mostly Not Satisfied,
2	Somewhat not Satisfied
3	Somewhat Satisfied
4	Mostly Satisfied
5	Completely Satisfied

The value boxed in red above represents one score relative to HUD's for one of the six iterations.

The value boxed in red below represents (one averaged score after 6 iterations relative to HUD's for one of the six iterations) * (Weighting of 0.1 for limited complexity).

TITLE: Summary Graph	Evaluation Matrix for Alternative Communication Methods	Limited Complexity of Technique	Easy Implementation	Programmable, or ability to input different routes	Reduce Chance of accidents	Limited line of sight movement	TOTALS
	WEIGHT	0.1	0.1	0.2	0.3	0.3	1
Alternative	Model	Requirement 1	Requirement 2	Requirement 3	Requirement 4	Requirement 5	Requirement Total
1	Heads Up Displays	0.3	0.3	0.6	1.2	1.4	3.8
2	Dash based GPS with Audio Directions	0.5	0.3	0.7	1.5	1.5	4.5
3	Phone based GPS with audio directions	0.5	0.4	0.7	1.1	1.1	3.8
4	Original Paper Maps	0.5	0.5	0.0	0.0	0.3	1.3

This final score is a sum of all the averaged and weighted scores for method 2.

0	Completely Not Satisfied
1	Mostly Not Satisfied,
2	Somewhat not Satisfied
3	Somewhat Satisfied
4	Mostly Satisfied
5	Completely Satisfied

Appendix D - Source Code Used to Compile Data Files from the City

```
import os
import zipfile

def FileConversion(Path):
    ReturnList = []
    DataFile = open(Path, "r")
    for Line in DataFile:
        LineList = Line.split(",")
        ReturnList.append(LineList)
    DataFile.close()
    return ReturnList

def ChaffRemoval(FileContents):
    ReturnList = []
    FirstRow = True
    Plow = False
    Sand = False
    Turn = False
    for Row in FileContents:
        if FirstRow:
            FirstRow = False
        else:
            if str(Row[12]) == '33':
                Plow = True
            if str(Row[12]) == '32':
                Plow = False
            if str(Row[12]) == '35':
                Sand = True
            if str(Row[12]) == '34':
                Sand = False
            if str(Row[12]) == '10':
                Turn = True
            if str(Row[12]) != '10':
                Turn = False
            Seconds = int(Row[5][(len(Row[5])-3)]) * 10 + int(Row[5][(len(Row[5])-2)])
            Minutes = int(Row[5][(len(Row[5]) - 6)]) * 10 + int(Row[5][(len(Row[5]) - 5)])
            Hours = int(Row[5][(len(Row[5]) - 9)]) * 10 + int(Row[5][(len(Row[5]) - 8)])
            Time = Hours * 3600 + Minutes * 60 + Seconds
```

```
            ReturnList.append([float(Row[0]), float(Row[1]), float(Row[3]), float(Row[4]), Time, Plow,
Sand, Turn])
    return ReturnList   #Lat, long, speed, heading, time, plowing, sanding, turning

def TurnFlagging(ReducedContents):
    TurnFlagged = ReducedContents
    Turning = False
    FlagCount = 1
    for Row in TurnFlagged:
        if Turning == False:
            if Row[7]:
                Turning = True
                Row[7] = FlagCount
            else:
                Row[7] = 0
        else:
            if (Row[7] == False):
                Turning = False
                Row[7] = FlagCount
                FlagCount += 1
            else:
                Row[7] = 0
    return TurnFlagged

def GenerateTurnTimeList(TurnFlagged):
    TurnCount = 0
    ReturnList = []
    for Row in TurnFlagged:
        if Row[7] > TurnCount:
            TurnCount = Row[7]
    for Turn in range(TurnCount):   #(Turn+1) is the number of turn
        #print(Turn)
        TurnTimestamps = []
        for Row in TurnFlagged:
            if Row[7] == (Turn + 1):
                TurnTimestamps.append(Row[4])
            #if len(TurnTimestamps) > 2:
            #print('SFY')
        if len(TurnTimestamps) > 1:
            ReturnList.append(TurnTimestamps[1] - TurnTimestamps[0])
    #print(TurnCount)
    return ReturnList
```

```
def WriteOutputFile(List, File):
    OutputFile = open(File, "w")
    for Item in List:
        OutputFile.write(str(Item) + "\n")
    return True

def ExtractSpeed(ReducedContents):
    ReturnList = []
    for Row in ReducedContents:
        if len(Row) > 2:
            ReturnList.append(Row[2])
    return ReturnList

def FindFileExtension(File):
    ReturnString = FileExtenstion = (File[len(File)-3] + File[len(File)-2] + File[len(File)-1])
    return ReturnString

def ProcessFolder(Path, TargetVehicle):
    ReturnTurnList = []
    ReturnSpeedList = []
    ReturnTallyInt = 0
    for FileName in os.listdir(Path):
        #print(FileName)
        #print(Vehicle)
        FilePath = Path + '\\' + FileName
        FileExtenstion = (FileName[len(FileName) - 3] + FileName[len(FileName) - 2] +
FileName[len(FileName) - 1])
        if FileExtenstion == "csv" and FileName != 'Turns.csv' and FileName != 'Speeds.csv':
            Vehicle = (FileName[4] + FileName[5] + FileName[6] + FileName[7])
            if Vehicle == TargetVehicle:
                #print(FileName)
                FileContents = FileConversion(FilePath)
                ReducedContents = ChaffRemoval(FileContents)
                TurnFlagged = TurnFlagging(ReducedContents)
                FileTurnList = GenerateTurnTimeList(TurnFlagged)
                for Turn in FileTurnList:
                    ReturnTurnList.append(Turn)
                FileSpeedList = ExtractSpeed(ReducedContents)
                for Speed in FileSpeedList:
                    ReturnSpeedList.append(Speed)
                print('Processed the following file:  ' + FileName)
                ReturnTallyInt = 1
    return ReturnTurnList, ReturnSpeedList, ReturnTallyInt
```

```
#-------------------------------------------------------------------------------------------------------------------
#-------------------------------------------------------------------------------------------------------------------
#This is where imports and definitions end, body of program begins
#-------------------------------------------------------------------------------------------------------------------
#-------------------------------------------------------------------------------------------------------------------


BasePath = input("Input target directory:  ")
OutputPath = input("Input output directory:  ")
print('\n')
#'C:\CapstoneData'

VehicleList = ['1806', '2682', '3028', '3110', '3313', '3626', '3727', '4272', '4273']

TurnList = []
SpeedList = []
AllTurnList = []
AllSpeedList = []

Tally = 0
UnzipTally = 0

n = 0
while n <= int(VehicleList[len(VehicleList)-1]):
   n += 1
   TurnList.append([])
   SpeedList.append([])

for BaseFolder in os.listdir(BasePath):
   if os.path.isdir(os.path.join(BasePath, BaseFolder)):
      SubPath = BasePath + '\\' + BaseFolder
      #print(SubPath)
      for SubFolder in os.listdir(SubPath):
         if os.path.isdir(os.path.join(SubPath, SubFolder)):
            #print(SubFolder)
            if SubFolder == "Tracks":
               DatePath = SubPath + '\\' + SubFolder
               #print(DatePath)
               for DateFolder in os.listdir(DatePath):
                  #print(DateFolder)
                  ZippedPath = DatePath + '\\' + DateFolder
                  #print(ZippedPath)
```

```
        for ZippedFile in os.listdir(ZippedPath):
            FileExtenstion = FindFileExtension(ZippedFile)
            if FileExtenstion == 'zip':
                FileToUnzipPath = ZippedPath + '\\' + ZippedFile
                with zipfile.ZipFile(FileToUnzipPath, 'r') as zip_ref:
                    zip_ref.extractall(ZippedPath)
                    zip_ref.close()
                    os.remove(FileToUnzipPath)
                print('Located and uncompressed the following file:  ' + ZippedFile)
                UnzipTally += 1
        for Vehicle in VehicleList:
            MyInt = 0
            VehicleTurnList = []
            VehicleSpeedList = []
            VehicleTurnList, VehicleSpeedList, MyInt = ProcessFolder(ZippedPath, Vehicle)
            Tally += MyInt
            for Turn in VehicleTurnList:
                TurnList[int(Vehicle)].append(Turn)
            for Speed in VehicleSpeedList:
                SpeedList[int(Vehicle)].append(Speed)
    elif SubFolder == "Bozeman":
        DatePath = SubPath + '\\' + SubFolder + '\\' + 'Tracks'
        #print(DatePath)
        for DateFolder in os.listdir(DatePath):
            #print(DateFolder)
            ZippedPath = DatePath + '\\' + DateFolder
            #print(ZippedPath)
            for ZippedFile in os.listdir(ZippedPath):
                FileExtenstion = FindFileExtension(ZippedFile)
                if FileExtenstion == 'zip':
                    FileToUnzipPath = ZippedPath + '\\' + ZippedFile
                    with zipfile.ZipFile(FileToUnzipPath, 'r') as zip_ref:
                        zip_ref.extractall(ZippedPath)
                        zip_ref.close()
                        os.remove(FileToUnzipPath)
                    print('Located and uncompressed the following file:  ' + ZippedFile)
                    UnzipTally += 1
            for Vehicle in VehicleList:
                MyInt = 0
                VehicleTurnList = []
                VehicleSpeedList = []
                VehicleTurnList, VehicleSpeedList, MyInt = ProcessFolder(ZippedPath, Vehicle)
                Tally += MyInt
```

```
                    for Turn in VehicleTurnList:
                        TurnList[int(Vehicle)].append(Turn)
                    for Speed in VehicleSpeedList:
                        SpeedList[int(Vehicle)].append(Speed)

for Vehicle in VehicleList:
    if len(TurnList[int(Vehicle)]) > 1:
        WriteOutputFile(TurnList[int(Vehicle)], (OutputPath +  '\\' + 'Turns' + Vehicle + '.csv'))
        print('Wrote to the following output file:  ' + 'Turns' + Vehicle + '.csv')
        for Turn in TurnList[int(Vehicle)]:
            AllTurnList.append(Turn)
    if len(SpeedList[int(Vehicle)]) > 1:
        WriteOutputFile(SpeedList[int(Vehicle)], (OutputPath + '\\' + 'Speed' + Vehicle + '.csv'))
        print('Wrote to the following output file:  ' + 'Speed' + Vehicle + '.csv')
        for Speed in SpeedList[int(Vehicle)]:
            AllSpeedList.append(Speed)

WriteOutputFile(AllTurnList, OutputPath +  '\\' + 'Turns.csv')
print('Wrote to the following output file:  ' + 'Turns.csv')
WriteOutputFile(AllSpeedList, OutputPath +  '\\' + 'Speed.csv')
print('Wrote to the following output file:  ' + 'Speed.csv')
print(str(UnzipTally) + " .zip's were uncompressed in total.")
print(str(Tally) + " .csv's were processed in total.")
m = input("Press enter to exit.")
```

Appendix E - List of Arcs Between Nodes

[[1,2,3],[2,1,4,5],[3,1,4,7],[4,2,3,5,8],[5,2,4],[6,9],[7,3,12,24],[8,4,9,13],[9,6,8,14],[10,34],[11,12,21],[12,11,7,23],[13,8,14],[14,9,18],[15,35],[16,20],[17,20,70],[18,14,28,29],[19,31,33],[20,16,17,36],[21,11,22],[22,21,23,52],[23,12,22,24,53],[24,7,23,25,54],[25,24,26,40],[26,13,25,27,37],[27,26,28,37],[28,28,27,29],[29,18,28,39,42],[30,31,38],[31,19,30,32],[32,31,33,38],[33,19,32,34],[34,10,33,35],[35,15,36],[36,20,35,43],[37,27,26,41],[38,30,32,49],[39,29,48,49],[40,25,41,55],[41,37,40,42,57],[42,29,41],[43,36,44,45],[44,43,45,62],[45,43,44],[46,47,58],[47,46,48,59],[48,39,47,60],[49,38,39,61],[50,51,73],[51,50,52,74],[52,22,51,75],[53,23,52,54,76],[54,24,53,55],[55,40,54,56,77],[56,55,57,67],[57,41,56,58,80],[58,46,57,59,81],[59,47,58,60],[60,48,59,61],[61,49,60,62,64],[62,44,61,63,270],[63,62],[64,61,66,69],[65,66],[66,64,65,68],[67,56,78,79],[68,66,69,71],[69,64,68,270,278],[70,17,88,270],[71,68,83,278],[72,73],[73,50,72,74],[74,51,73,75,272],[75,52,74,76],[76,53,75,77,147],[77,55,76,78],[78,67,77,79],[79,67,78,80],[80,57,79,81,273],[81,58,80,82,100],[82,81,83,93],[83,71,82,84,89],[84,83,85,90],[85,84,86,91,278],[86,85,87,91],[87,86,271,97],[88,70,271,116,131],[89,83,90,94],[90,84,89,91,95],[91,85,86,90,96],[92,272],[93,82,102,94],[94,89,93,95,103],[95,90,94,96],[96,91,95,97,107],[97,87,96,110],[98,78,99,279],[99,98,132,273],[100,81,101,134,273],[101,100],[102,93,103,274],[103,94,275,102],[104,274,105,118],[105,104,106,119],[106,105,275,120],[107,96,275,108,121],[108,107,109,122],[109,108,110,123],[110,97,109,111,124],[111,112,110,125],[112,111,113,126],[113,112,114,127],[114,113,115,128,271],[115,114,116,129],[116,88,115,130],[117,118,135,274],[118,104,117,137],[119,105,118,139],[120,106,119,141],[121,107,120,158],[122,108,121,160],[123,109,122,161],[124,110,123,162],[125,111,124,163],[126,112,125,164],[127,113,126,165],[128,114,127,129,166],[129,115,128,130,167],[130,116,129,131,142],[131,88,130,143],[132,99,133,152],[133,273,132,134,153],[134,100,133,135],[135,117,134,136,154],[136,135,137,155],[137,136,118,138],[138,137,139,156],[139,119,138,140],[140,139,141,157],[141,120,140],[142,130,143,168],[143,131,142,169],[144,145],[145,144,146,187],[146,272,145,147,187],[147,76,146,148,276],[148,147,279,189],[149,279,152,150,151],[150,151,149,192],[151,149,193],[152,149,132,153],[153,133,152,154,194],[154,153,135,171,155],[155,136,154,156,172],[156,138,155,157,173],[157,156,158,140,174],[158,121,159],[159,160,175],[160,122,161],[161,123,162,176],[162,124,163,177],[163,125,164,178],[164,126,165,179],[165,127,166,180],[166,128,167,181],[167,165,168,129,182],[168,142,183],[169,143,170,207],[170,169,208],[171,154,172,197],[172,155,171,173],[173,156,172,174],[174,157,173,175,198],[175,159,174,176],[176,161,175,177],[177,162,176,178,202],[178,163,177,179],[179,164,178,180],[180,165,179,181],[181,166,180,182],[182,167,181,183,184],[183,168,182,185],[184,182,185,206],[185,183,184,186],[186,185],[187,145,146,187,209,276],[188,189,276],[189,148,188,211],[190,191],[191,190,192,211],[192,150,191,212],[193,151,194,213],[194,193,153,195,196],[195,194,214,215],[196,194,197,216],[197,171,196,198,217],[198,174,197,199],[199,198,200,201],[200,199,201],[201,199,200,218],[202,177,203,204],[203,202,205,219],[204,202,205,206],[205,203,204,220],[206,184,204,252],[207,169,237,208,236],[208,170,207,255],[209,187,210,238],[210,276,209,211],[211,189,230],[212,191,192,213],[213,193,212,214,222],[214,195,213,215,231],[215,214,195,216],[216,196,215,217],[217,197,216,218],[218,201,217,219,224],[219,203,218,220,226],[220,205,219,221,227],[221,220],[222,213,223,230],[223,222],[224,218,225],[225,224,228,281],[226,219,227,228],[227,220,226,229],[228,225,226,229,232],[229,227,228,233],[230,211,222,223,

231],[231,214,230,239],[232,233,228,231,235],[233,229,232,234],[234,233],[235,232,251,231],[236,207,237,280],[237,207,236],[238,209],[239,231,240,243],[240,239,241],[241,240,250],[242,243,246,247],[243,244,239,242,248],[244,243],[245,246],[246,245,242,247],[247,242,246,248],[248,243,247,249,263],[249,240,248,250],[250,241,249,251,256],[251,235,250,252,259],[252,206,251,255],[253,280,254,255],[254,280,253],[255,208,252,253],[256,250,257,264],[257,256],[258,259],[259,251,258,260],[260,259,277],[261,262],[262,261,263],[263,248,262,267],[264,256,263,265],[265,264,266,268],[266,259,265,277],[267,263],[268,265],[269,277],[270,62,69,70,271],[271,270,87,88,114],[272,92,74,146],[273,80,99,100,133],[274,102,104,117],[275,103,106,107],[276,147,187,188,210],[277,260,266,269],[278,69,71,85],[279,98,148,149],[280,236,253,254]]

Appendix F - Arc Cost Matrix

[Node 1, Node 2, Distance (meters)]

[[1, 2, 871], [1, 3, 1130], [2, 4, 599], [2, 5, 701], [3, 4, 667], [3, 7, 808], [4, 5, 420], [4, 8, 1360], [6, 9, 357], [7, 12, 1100], [7, 24, 805], [8, 9, 284], [8, 13, 362], [9, 14, 366], [10, 34, 1160], [11, 12, 805], [11, 21, 556], [12, 23, 500], [13, 14, 280], [14, 18, 217], [15, 35, 331], [16, 20, 300], [17, 20, 606], [17, 70, 1960], [18, 28, 256], [18, 29, 351], [19, 31, 201], [19, 33, 310], [20, 36, 250], [21, 22, 402], [22, 23, 403], [22, 52, 800], [23, 24, 810], [23, 53, 817], [24, 25, 800], [24, 54, 796], [25, 26, 553], [25, 40, 398], [26, 27, 131], [26, 37, 230], [27, 28, 130], [27, 37, 226], [28, 29, 174], [29, 39, 751], [29, 42, 392], [30, 31, 528], [30, 38, 430], [31, 32, 76], [32, 33, 34], [32, 38, 542], [33, 34, 387], [34, 35, 301], [35, 36, 200], [36, 43, 440], [37, 41, 210], [38, 49, 515], [39, 48, 271], [39, 49, 553], [40, 41, 763], [40, 55, 410], [41, 42, 269], [41, 57, 392], [43, 44, 173], [43, 45, 345], [44, 45, 214], [44, 62, 372], [46, 47, 142], [46, 58, 247], [47, 48, 277], [47, 59, 236], [48, 60, 234], [49, 61, 148], [50, 51, 801], [50, 73, 785], [51, 52, 420], [51, 74, 783], [52, 75, 808], [53, 54, 792], [53, 76, 800], [54, 55, 787], [55, 56, 157], [55, 77, 812], [56, 57, 654], [56, 67, 429], [57, 58, 428], [57, 80, 819], [58, 59, 148], [58, 81, 834], [59, 60, 283], [60, 61, 397], [61, 62, 1180], [61, 64, 277], [62, 63, 80], [62, 270, 327], [64, 66, 114], [64, 69, 391], [65, 66, 96], [66, 68, 110], [67, 78, 370], [67, 79, 672], [68, 69, 180], [68, 71, 113], [69, 270, 956], [69, 278, 108], [70, 88, 331], [70, 270, 249], [71, 83, 226], [71, 278, 177], [72, 73, 417], [73, 74, 793], [74, 75, 411], [74, 272, 197], [75, 76, 401], [76, 77, 1630], [76, 147, 813], [77, 78, 213], [78, 79, 339], [79, 80, 274], [80, 81, 406], [80, 273, 400], [81, 82, 422], [81, 100, 395], [82, 83, 397], [82, 93, 288], [83, 84, 95], [83, 89, 105], [84, 85, 84], [84, 90, 105], [85, 86, 93], [85, 91, 107], [85, 278, 233], [86, 87, 313], [86, 91, 199], [87, 271, 544], [87, 97, 278], [88, 271, 256], [88, 116, 466], [88, 131, 886], [89, 90, 93], [89, 94, 172], [90, 91, 181], [90, 95, 165], [91, 96, 169], [92, 272, 315], [93, 102, 111], [93, 94, 401], [94, 95, 95], [94, 103, 111], [95, 96, 88], [96, 97, 401], [96, 107, 191], [97, 110, 182], [98, 279, 431], [98, 99, 507], [99, 132, 371], [99, 273, 96], [100, 101, 544], [100, 134, 312], [100, 273, 407], [102, 103, 404], [102, 274, 80], [103, 275, 75], [104, 274, 81], [104, 105, 111], [104, 118, 110], [105, 106, 11], [105, 119, 109], [106, 275, 100], [106, 120, 107], [107, 275, 188], [107, 108, 194], [107, 121, 107], [108, 109, 101], [108, 122, 110], [109, 110, 99], [109, 123, 109], [110, 111, 116], [110, 124, 108], [111, 112, 130], [111, 125, 111], [112, 113, 110], [112, 126, 120], [113, 114, 182], [113, 127, 115], [114, 115, 131], [114, 128, 116], [114, 271, 462], [115, 116, 135], [115, 129, 118], [116, 130, 116], [117, 118, 82], [117, 135, 114], [117, 274, 114], [118, 119, 110], [118, 137, 115], [119, 120, 110], [119, 139, 115], [120, 121, 288], [120, 141, 115], [121, 122, 193], [121, 158, 114], [122, 123, 102], [122, 160, 114], [123, 124, 99], [123, 161, 114], [124, 125, 116], [124, 162, 121], [125, 126, 114], [125, 163, 121], [126, 127, 110], [126, 164, 59], [127, 128, 181], [127, 165, 123], [128, 129, 133], [128, 166, 122], [129, 130, 134], [129, 167, 122], [130, 131, 229], [130, 142, 123], [131, 143, 106], [132, 133, 184], [132, 152, 196], [133, 273, 322], [133, 134, 425], [133, 153, 94], [134, 135, 427], [135, 136, 69], [135, 154, 109], [136, 137, 27], [136, 155, 108], [137, 138, 85], [138, 139, 25], [138, 156, 106], [139, 140, 77], [140, 141, 32], [140, 157, 150], [142, 143, 195], [142, 168, 108], [143, 169, 606], [144, 145, 988], [145, 146, 171], [145, 187, 522], [146, 272, 1230], [146, 147, 1630], [146, 187, 395], [147, 148, 810], [147,

276, 590], [148, 279, 956], [148, 189, 682], [149, 152, 99], [149, 150, 72], [149, 151, 119], [149, 279, 320], [150, 151, 84], [150, 192, 426], [151, 193, 299], [152, 153, 272], [153, 154, 839], [153, 194, 349], [154, 171, 118], [154, 155, 84], [155, 156, 110], [155, 172, 55], [156, 157, 103], [156, 173, 53], [157, 158, 317], [157, 174, 117], [158, 159, 117], [159, 160, 77], [159, 175, 117], [160, 161, 99], [161, 162, 107], [161, 176, 110], [162, 163, 118], [162, 177, 110], [163, 164, 117], [163, 178, 108], [164, 165, 110], [164, 179, 107], [165, 166, 176], [165, 180, 108], [166, 167, 137], [166, 181, 112], [167, 168, 135], [167, 182, 112], [168, 183, 110], [169, 170, 169], [169, 207, 777], [170, 208, 883], [171, 172, 92], [171, 197, 225], [172, 173, 107], [173, 174, 102], [174, 175, 500], [174, 198, 222], [175, 176, 107], [176, 177, 107], [177, 178, 121], [177, 202, 328], [178, 179, 112], [179, 180, 109], [180, 181, 173], [181, 182, 138], [182, 183, 138], [182, 184, 108], [183, 185, 109], [184, 185, 141], [184, 206, 273], [185, 186, 110], [187, 209, 407], [187, 276, 880], [188, 189, 307], [188, 276, 804], [189, 211, 122], [190, 191, 77], [191, 192, 541], [191, 211, 547], [192, 212, 181], [193, 194, 400], [193, 213, 381], [194, 195, 217], [194, 196, 426], [195, 214, 252], [195, 215, 495], [196, 197, 408], [196, 216, 462], [197, 198, 304], [197, 217, 467], [198, 199, 202], [199, 200, 210], [199, 201, 111], [200, 201, 102], [201, 218, 127], [202, 203, 109], [202, 204, 160], [203, 205, 120], [203, 219, 233], [204, 205, 69], [204, 206, 537], [205, 220, 234], [206, 252, 1970], [207, 237, 337], [207, 208, 631], [207, 236, 723], [208, 255, 3690], [209, 210, 814], [209, 238, 834], [210, 276, 199], [210, 211, 809], [211, 230, 1680], [212, 213, 266], [213, 214, 375], [213, 222, 339], [214, 215, 323], [214, 231, 392], [215, 216, 91], [216, 217, 408], [217, 218, 309], [218, 219, 708], [218, 224, 226], [219, 220, 117], [219, 226, 120], [220, 221, 134], [220, 227, 121], [222, 223, 109], [222, 230, 114], [224, 225, 201], [225, 231, 107], [225, 228, 505], [226, 227, 122], [226, 228, 109], [227, 229, 109], [228, 229, 120], [228, 232, 222], [229, 233, 224], [230, 231, 276], [231, 239, 403], [232, 233, 95], [232, 235, 220], [233, 234, 399], [235, 251, 581], [236, 237, 626], [236, 242, 369], [239, 240, 420], [239, 243, 203], [240, 241, 412], [241, 250, 412], [242, 243, 111], [242, 246, 340], [242, 247, 217], [243, 244, 52], [243, 248, 221], [245, 246, 130], [246, 247, 125], [247, 248, 110], [248, 249, 156], [248, 263, 1200], [249, 250, 465], [250, 251, 841], [250, 256, 532], [251, 252, 1330], [251, 259, 858], [252, 255, 360], [253, 254, 624], [253, 255, 91], [256, 257, 229], [256, 264, 660], [258, 259, 547], [259, 260, 169], [260, 277, 276], [261, 262, 278], [262, 263, 500], [263, 267, 817], [264, 265, 491], [265, 266, 402], [265, 268, 1610], [266, 277, 156], [269, 277, 1210], [270, 271, 332], [280, 253, 737], [280, 254, 681]]

Appendix G - Turn Penalty Matrix

[[1, 2, 4], [4, 2, 1], [1, 3, 4], [4, 3, 1], [2, 1, 3], [3, 1, 2], [2, 4, 3], [3, 4, 2], [2, 4, 5], [5, 4, 2], [3, 4, 8], [8, 4, 3], [3, 7, 12], [12, 7, 3], [4, 8, 13], [13, 8, 4], [5, 4, 8], [8, 4, 5], [6, 9, 8], [8, 9, 6], [7, 12, 11], [11, 12, 7], [7, 24, 23], [23, 24, 7], [7, 24, 25], [25, 24, 7], [8, 13, 14], [14, 13, 8], [8, 9, 14], [14, 9, 8], [9, 8, 13], [13, 8, 9], [9, 14, 13], [13, 14, 9], [10, 34, 33], [33, 34, 10], [10, 34, 35], [35, 34, 10], [11, 21, 22], [22, 21, 11], [11, 12, 23], [23, 12, 11], [12, 23, 22], [22, 23, 12], [12, 23, 24], [24, 23, 12], [12, 7, 24], [24, 7, 12], [13, 14, 18], [18, 14, 13], [14, 18, 29], [29, 18, 14], [15, 35, 34], [34, 35, 15], [15, 35, 36], [36, 35, 15], [16, 20, 17], [17, 20, 16], [16, 20, 36], [36, 20, 16], [17, 70, 270], [270, 70, 17], [18, 29, 28], [28, 29, 18], [18, 28, 27], [27, 28, 18], [19, 31, 30], [30, 31, 19], [19, 31, 32], [32, 31, 19], [19, 33, 32], [32, 33, 19], [19, 33, 34], [34, 33, 19], [20, 17, 70], [70, 17, 20], [20, 36, 35], [35, 36, 20], [21, 22, 52], [52, 22, 21], [22, 23, 53], [53, 23, 22], [22, 52, 51], [51, 52, 22], [23, 24, 54], [54, 24, 23], [24, 23, 53], [53, 23, 24], [24, 25, 40], [40, 25, 24], [24, 54, 53], [53, 54, 24], [25, 24, 54], [54, 24, 25], [25, 40, 41], [41, 40, 25], [25, 26, 37], [37, 26, 25], [26, 27, 37], [37, 27, 26], [26, 37, 27], [27, 37, 26], [29, 42, 41], [41, 42, 29], [29, 39, 48], [48, 39, 29], [30, 38, 32], [32, 38, 30], [31, 32, 38], [38, 32, 31], [31, 30, 38], [38, 30, 31], [31, 19, 33], [33, 19, 31], [35, 36, 43], [43, 36, 35], [36, 43, 45], [45, 43, 36], [37, 41, 40], [40, 41, 37], [37, 41, 42], [42, 41, 37], [38, 49, 39], [39, 49, 38], [39, 48, 47], [47, 48, 39], [39, 49, 61], [61, 49, 39], [40, 55, 54], [54, 55, 40], [40, 55, 56], [56, 55, 40], [40, 41, 57], [57, 41, 40], [41, 57, 56], [56, 57, 41], [41, 57, 58], [58, 57, 41], [42, 41, 57], [57, 41, 42], [43, 44, 45], [45, 44, 43], [43, 45, 44], [44, 45, 43], [45, 44, 62], [62, 44, 45], [44, 43, 45], [45, 43, 44], [44, 62, 63], [63, 62, 44], [44, 62, 61], [61, 62, 44], [46, 47, 59], [59, 47, 46], [46, 58, 57], [57, 58, 46], [46, 58, 59], [59, 58, 46], [47, 46, 58], [58, 46, 47], [47, 48, 60], [60, 48, 47], [47, 59, 58], [58, 59, 47], [47, 59, 60], [60, 59, 47], [48, 47, 59], [59, 47, 48], [48, 39, 49], [49, 39, 48], [48, 60, 59], [59, 60, 48], [48, 60, 61], [61, 60, 48], [49, 61, 60], [60, 61, 49], [49, 61, 62], [62, 61, 49], [50, 51, 74], [74, 51, 50], [50, 73, 72], [72, 73, 50], [50, 73, 74], [74, 73, 50], [51, 52, 75], [75, 52, 51], [51, 74, 73], [73, 74, 51], [51, 74, 75], [75, 74, 51], [51, 50, 73], [73, 50, 51], [52, 75, 76], [76, 75, 52], [52, 75, 74], [74, 75, 52], [52, 51, 74], [74, 51, 52], [54, 53, 76], [76, 53, 54], [54, 55, 77], [77, 55, 54], [56, 55, 77], [77, 55, 56], [56, 67, 79], [79, 67, 56], [56, 57, 80], [80, 57, 56], [57, 56, 67], [67, 56, 57], [57, 58, 81], [81, 58, 57], [57, 80, 79], [79, 80, 57], [57, 80, 81], [81, 80, 57], [58, 57, 80], [80, 57, 58], [58, 81, 80], [80, 81, 58], [58, 81, 82], [82, 81, 58], [59, 58, 81], [81, 58, 59], [60, 61, 64], [64, 61, 60], [61, 62, 270], [270, 62, 61], [61, 64, 69], [69, 64, 61], [62, 61, 64], [64, 61, 62], [62, 270, 69], [69, 270, 62], [62, 270, 70], [70, 270, 62], [63, 62, 270], [270, 62, 63], [64, 66, 65], [65, 66, 64], [64, 69, 68], [68, 69, 64], [64, 69, 270], [270, 69, 64], [65, 66, 68], [68, 66, 65], [66, 64, 69], [69, 64, 66], [66, 68, 69], [69, 68, 66], [67, 78, 77], [77, 78, 67], [67, 78, 79], [79, 78, 67], [67, 79, 78], [78, 79, 67], [67, 79, 80], [80, 79, 67], [68, 71, 278], [278, 71, 68], [68, 69, 278], [278, 69, 68], [69, 68, 71], [71, 68, 69], [69, 278, 71], [71, 278, 69], [70, 270, 271], [271, 270, 70], [70, 88, 271], [271, 88, 70], [70, 88, 131], [131, 88, 70], [71, 278, 85], [85, 278, 71], [71, 83, 82], [82, 83, 71], [71, 83, 84], [84, 83, 71], [73, 74, 272], [272, 74, 73], [74, 272, 92], [92, 272, 74], [75, 74, 272], [272, 74, 75], [75, 76, 147], [147, 76, 75], [76, 147, 146], [146, 147, 76], [76, 147, 148], [148, 147, 76], [77, 76, 147], [147, 76, 77], [78, 67, 79], [79, 67, 78], [79, 80, 273], [273, 80, 79], [80, 273, 99], [99, 273, 80], [80, 273, 100], [100, 273, 80], [80, 81, 100], [100, 81, 80], [81, 100, 273], [273, 100, 81], [81, 100, 101], [101, 100,

81], [81, 82, 93], [93, 82, 81], [82, 81, 100], [100, 81, 82], [82, 83, 89], [89, 83, 82], [82, 93, 94], [94, 93, 82], [83, 82, 93], [93, 82, 83], [83, 84, 90], [90, 84, 83], [83, 89, 90], [90, 89, 83], [84, 83, 89], [89, 83, 84], [84, 85, 91], [91, 85, 84], [84, 90, 89], [89, 90, 84], [84, 90, 91], [91, 90, 84], [85, 86, 91], [91, 86, 85], [85, 84, 90], [90, 84, 85], [85, 91, 86], [86, 91, 85], [85, 91, 90], [90, 91, 85], [86, 87, 97], [97, 87, 86], [86, 85, 278], [278, 85, 86], [86, 85, 91], [91, 85, 86], [87, 86, 91], [91, 86, 87], [87, 271, 270], [270, 271, 87], [87, 271, 114], [114, 271, 87], [88, 271, 270], [270, 271, 88], [88, 70, 270], [270, 70, 88], [88, 131, 130], [130, 131, 88], [88, 116, 115], [115, 116, 88], [89, 94, 93], [93, 94, 89], [89, 94, 95], [95, 94, 89], [89, 90, 95], [95, 90, 89], [90, 89, 94], [94, 89, 90], [90, 95, 94], [94, 95, 90], [90, 95, 96], [96, 95, 90], [90, 91, 96], [96, 91, 90], [91, 90, 95], [95, 90, 91], [91, 96, 97], [97, 96, 91], [91, 96, 95], [95, 96, 91], [92, 272, 146], [146, 272, 92], [93, 94, 103], [103, 94, 93], [93, 102, 103], [103, 102, 93], [94, 93, 102], [102, 93, 94], [94, 103, 102], [102, 103, 94], [95, 96, 107], [107, 96, 95], [96, 97, 110], [110, 97, 96], [97, 96, 107], [107, 96, 97], [97, 110, 109], [109, 110, 97], [97, 110, 111], [111, 110, 97], [98, 99, 132], [132, 99, 98], [99, 132, 152], [152, 132, 99], [99, 132, 133], [133, 132, 99], [99, 273, 133], [133, 273, 99], [100, 273, 133], [133, 273, 100], [100, 134, 133], [133, 134, 100], [100, 134, 135], [135, 134, 100], [101, 100, 134], [134, 100, 101], [102, 103, 275], [275, 103, 102], [102, 274, 104], [104, 274, 102], [103, 102, 274], [274, 102, 103], [103, 275, 106], [106, 275, 103], [103, 275, 107], [107, 275, 103], [104, 274, 117], [117, 274, 104], [104, 118, 117], [117, 118, 104], [104, 105, 119], [119, 105, 104], [105, 104, 118], [118, 104, 105], [105, 106, 120], [120, 106, 105], [106, 105, 119], [119, 105, 106], [107, 108, 122], [122, 108, 107], [108, 107, 121], [121, 107, 108], [108, 109, 123], [123, 109, 108], [109, 108, 122], [122, 108, 109], [110, 109, 123], [123, 109, 110], [110, 111, 125], [125, 111, 110], [111, 110, 124], [124, 110, 111], [111, 112, 126], [126, 112, 111], [112, 111, 125], [125, 111, 112], [112, 113, 127], [127, 113, 112], [113, 112, 126], [126, 112, 113], [113, 114, 128], [128, 114, 113], [114, 113, 127], [127, 113, 114], [114, 128, 129], [129, 128, 114], [114, 115, 129], [129, 115, 114], [115, 114, 128], [128, 114, 115], [115, 129, 128], [128, 129, 115], [115, 129, 130], [130, 129, 115], [115, 116, 130], [130, 116, 115], [116, 115, 129], [129, 115, 116], [116, 130, 129], [129, 130, 116], [116, 130, 131], [131, 130, 116], [117, 135, 136], [136, 135, 117], [118, 137, 136], [136, 137, 118], [118, 117, 135], [135, 117, 118], [118, 137, 138], [138, 137, 118], [119, 139, 138], [138, 139, 119], [119, 139, 140], [140, 139, 119], [120, 141, 140], [140, 141, 120], [128, 129, 167], [167, 129, 128], [129, 128, 166], [166, 128, 129], [129, 130, 142], [142, 130, 129], [130, 129, 167], [167, 129, 130], [130, 142, 143], [143, 142, 130], [130, 131, 143], [143, 131, 130], [131, 130, 142], [142, 130, 131], [131, 143, 169], [169, 143, 131], [131, 143, 142], [142, 143, 131], [132, 99, 273], [273, 99, 132], [132, 133, 273], [273, 133, 132], [132, 133, 153], [153, 133, 132], [132, 152, 153], [153, 152, 132], [133, 153, 154], [154, 153, 133], [134, 133, 153], [153, 133, 134], [134, 135, 154], [154, 135, 134], [135, 136, 155], [155, 136, 135], [135, 154, 153], [153, 154, 135], [135, 154, 155], [155, 154, 135], [136, 155, 154], [154, 155, 136], [136, 135, 154], [154, 135, 136], [136, 155, 156], [156, 155, 136], [137, 136, 155], [155, 136, 137], [137, 138, 156], [156, 138, 137], [138, 156, 155], [155, 156, 138], [138, 156, 157], [157, 156, 138], [139, 138, 156], [156, 138, 139], [139, 140, 157], [157, 140, 139], [140, 157, 156], [156, 157, 140], [141, 140, 157], [157, 140, 141], [143, 142, 168], [168, 142, 143], [143, 169, 207], [207, 169, 143], [144, 145, 187], [187, 145, 144], [145, 146, 187], [187, 146, 145], [145, 187, 146], [146, 187, 145], [145, 187, 209], [209, 187, 145], [146, 145, 187], [187, 145, 146], [146, 187, 276], [276, 187, 146], [146, 147, 276], [276,

147, 146], [147, 276, 187], [187, 276, 147], [147, 148, 189], [189, 148, 147], [147, 276, 188], [188, 276, 147], [148, 189, 188], [188, 189, 148], [148, 147, 276], [276, 147, 148], [149, 152, 153], [153, 152, 149], [150, 192, 212], [212, 192, 150], [150, 149, 279], [279, 149, 150], [150, 149, 151], [151, 149, 150], [151, 149, 152], [152, 149, 151], [151, 193, 194], [194, 193, 151], [152, 153, 194], [194, 153, 152], [152, 149, 279], [279, 149, 152], [153, 194, 193], [193, 194, 153], [153, 194, 196], [196, 194, 153], [153, 154, 171], [171, 154, 153], [154, 153, 194], [194, 153, 154], [154, 171, 172], [172, 171, 154], [154, 155, 172], [172, 155, 154], [155, 154, 171], [171, 154, 155], [155, 172, 171], [171, 172, 155], [155, 172, 173], [173, 172, 155], [155, 156, 173], [173, 156, 155], [156, 155, 172], [172, 155, 156], [156, 173, 172], [172, 173, 156], [156, 173, 174], [174, 173, 156], [156, 157, 174], [174, 157, 156], [157, 174, 173], [173, 174, 157], [157, 156, 173], [173, 156, 157], [157, 174, 175], [175, 174, 157], [159, 175, 176], [176, 175, 159], [161, 176, 175], [175, 176, 161], [161, 176, 177], [177, 176, 161], [162, 177, 176], [176, 177, 162], [162, 177, 178], [178, 177, 162], [163, 178, 177], [177, 178, 163], [163, 178, 179], [179, 178, 163], [164, 179, 178], [178, 179, 164], [164, 179, 180], [180, 179, 164], [165, 180, 179], [179, 180, 165], [165, 180, 181], [181, 180, 165], [166, 181, 180], [180, 181, 166], [166, 181, 182], [182, 181, 166], [167, 182, 181], [181, 182, 167], [167, 182, 183], [183, 182, 167], [168, 183, 182], [182, 183, 168], [169, 170, 208], [208, 170, 169], [169, 207, 237], [237, 207, 169], [169, 207, 208], [208, 207, 169], [170, 169, 207], [207, 169, 170], [170, 208, 207], [207, 208, 170], [171, 197, 196], [196, 197, 171], [171, 197, 198], [198, 197, 171], [173, 174, 198], [198, 174, 173], [174, 198, 197], [197, 198, 174], [175, 174, 198], [198, 174, 175], [176, 177, 202], [202, 177, 176], [177, 202, 204], [204, 202, 177], [178, 177, 202], [202, 177, 178], [181, 182, 184], [184, 182, 181], [182, 183, 185], [185, 183, 182], [182, 184, 185], [185, 184, 182], [183, 182, 184], [184, 182, 183], [183, 185, 184], [184, 185, 183], [184, 185, 186], [186, 185, 184], [184, 206, 204], [204, 206, 184], [185, 184, 206], [206, 184, 185], [187, 209, 210], [210, 209, 187], [187, 276, 210], [210, 276, 187], [188, 276, 210], [210, 276, 188], [188, 189, 211], [211, 189, 188], [189, 188, 276], [276, 188, 189], [189, 148, 279], [279, 148, 189], [190, 191, 192], [192, 191, 190], [191, 192, 212], [212, 192, 191], [192, 212, 213], [213, 212, 192], [193, 194, 195], [195, 194, 193], [193, 213, 212], [212, 213, 193], [193, 213, 214], [214, 213, 193], [195, 214, 213], [213, 214, 195], [195, 214, 215], [215, 214, 195], [195, 194, 196], [196, 194, 195], [196, 216, 215], [215, 216, 196], [196, 216, 217], [217, 216, 196], [196, 197, 217], [217, 197, 196], [197, 196, 216], [216, 196, 197], [197, 217, 216], [216, 217, 197], [197, 217, 218], [218, 217, 197], [197, 198, 199], [199, 198, 197], [198, 197, 217], [217, 197, 198], [198, 199, 200], [200, 199, 198], [199, 200, 201], [201, 200, 199], [200, 201, 218], [218, 201, 200], [201, 218, 217], [217, 218, 201], [201, 218, 219], [219, 218, 201], [202, 204, 205], [205, 204, 202], [202, 204, 206], [206, 204, 202], [202, 203, 205], [205, 203, 202], [203, 202, 204], [204, 202, 203], [203, 205, 204], [204, 205, 203], [203, 219, 218], [218, 219, 203], [203, 219, 220], [220, 219, 203], [203, 205, 220], [220, 205, 203], [204, 206, 252], [252, 206, 204], [205, 204, 206], [206, 204, 205], [205, 203, 219], [219, 203, 205], [205, 220, 219], [219, 220, 205], [205, 220, 221], [221, 220, 205], [206, 252, 251], [251, 252, 206], [206, 252, 255], [255, 252, 206], [207, 237, 236], [236, 237, 207], [207, 236, 237], [237, 236, 207], [207, 208, 255], [255, 208, 207], [208, 207, 236], [236, 207, 208], [208, 255, 253], [253, 255, 208], [209, 187, 276], [276, 187, 209], [209, 210, 276], [276, 210, 209], [210, 209, 238], [238, 209, 210], [211, 230, 222], [222, 230, 211], [212, 213, 222], [222, 213, 212], [213, 222, 223], [223, 222, 213], [213, 214, 231],

[231, 214, 213], [214, 213, 222], [222, 213, 214], [214, 231, 230], [230, 231, 214], [214, 195, 215], [215, 195, 214], [217, 218, 224], [224, 218, 217], [218, 224, 225], [225, 224, 218], [218, 219, 226], [226, 219, 218], [219, 218, 224], [224, 218, 219], [219, 226, 227], [227, 226, 219], [219, 220, 227], [227, 220, 219], [220, 227, 226], [226, 227, 220], [220, 219, 226], [226, 219, 220], [221, 220, 227], [227, 220, 221], [222, 230, 231], [231, 230, 222], [223, 222, 230], [230, 222, 223], [224, 225, 281], [281, 225, 224], [225, 281, 232], [232, 281, 225], [225, 228, 232], [232, 228, 225], [226, 227, 229], [229, 227, 226], [226, 228, 229], [229, 228, 226], [227, 229, 228], [228, 229, 227], [228, 229, 233], [233, 229, 228], [228, 232, 233], [233, 232, 228], [228, 225, 281], [281, 225, 228], [228, 232, 281], [281, 232, 228], [229, 228, 232], [232, 228, 229], [229, 233, 232], [232, 233, 229], [229, 233, 234], [234, 233, 229], [230, 231, 239], [239, 231, 230], [231, 239, 240], [240, 239, 231], [232, 281, 235], [235, 281, 232], [233, 232, 235], [235, 232, 233], [235, 251, 250], [250, 251, 235], [235, 251, 252], [252, 251, 235], [236, 280, 254], [254, 280, 236], [236, 207, 237], [237, 207, 236], [237, 236, 280], [280, 236, 237], [239, 243, 244], [244, 243, 239], [239, 243, 242], [242, 243, 239], [240, 239, 243], [243, 239, 240], [240, 241, 250], [250, 241, 240], [242, 246, 245], [245, 246, 242], [242, 246, 247], [247, 246, 242], [242, 247, 248], [248, 247, 242], [242, 243, 248], [248, 243, 242], [243, 248, 247], [247, 248, 243], [243, 248, 249], [249, 248, 243], [244, 243, 248], [248, 243, 244], [246, 242, 247], [247, 242, 246], [247, 248, 263], [263, 248, 247], [248, 263, 262], [262, 263, 248], [249, 250, 256], [256, 250, 249], [250, 251, 259], [259, 251, 250], [250, 256, 257], [257, 256, 250], [251, 259, 260], [260, 259, 251], [251, 259, 258], [258, 259, 251], [252, 251, 259], [259, 251, 252], [252, 255, 253], [253, 255, 252], [253, 254, 280], [280, 254, 253], [253, 280, 254], [254, 280, 253], [254, 253, 255], [255, 253, 254], [256, 264, 265], [265, 264, 256], [257, 256, 264], [264, 256, 257], [260, 277, 266], [266, 277, 260], [260, 277, 269], [269, 277, 260], [262, 263, 267], [267, 263, 262], [264, 265, 268], [268, 265, 264], [266, 277, 269], [269, 277, 266], [266, 265, 268], [268, 265, 266]]

Appendix H - Source Code Used to Generate Turn Penalty Terms, Arc List, and Cost Matrix

```
#-----------------------------------------------------------------------------------------------------------------------
#Variables
#-----------------------------------------------------------------------------------------------------------------------

NodeList =
[[1,2,3],[2,1,4,5],[3,1,4,7],[4,2,3,5,8],[5,2,4],[6,9],[7,3,12,24],[8,4,9,13],[9,6,8,14],[10,34],[11,12,21],[12,
11,7,23],[13,8,14],[14,9,13,18],[15,35],[16,20],[17,20,70],[18,14,28,29],[19,31,33],[20,16,17,36],[21,11,2
2],[22,21,23,52],[23,12,22,24,53],[24,7,23,25,54],[25,24,26,40],[26,13,25,27,37],[27,26,28,37],[28,18,27,
29],[29,18,28,39,42],[30,31,38],[31,19,30,32],[32,31,33,38],[33,19,32,34],[34,10,33,35],[35,15,34,36],[3
6,20,35,43],[37,27,26,41],[38,30,32,49],[39,29,48,49],[40,25,41,55],[41,37,40,42,57],[42,29,41],[43,36,4
4,45],[44,43,45,62],[45,43,44],[46,47,58],[47,46,48,59],[48,39,47,60],[49,38,39,61],[50,51,73],[51,50,52,
74],[52,22,51,75],[53,23,52,54,76],[54,24,53,55],[55,40,54,56,77],[56,55,57,67],[57,41,56,58,80],[58,46,
57,59,81],[59,47,58,60],[60,48,59,61],[61,49,60,62,64],[62,44,61,63,270],[63,62],[64,61,66,69],[65,66],[
66,64,65,68],[67,56,78,79],[68,66,69,71],[69,64,68,270,278],[70,17,88,270],[71,68,83,278],[72,73],[73,5
0,72,74],[74,51,73,75,272],[75,52,74,76],[76,53,75,77,147],[77,55,76,78],[78,67,77,79],[79,67,78,80],[80
,57,79,81,273],[81,58,80,82,100],[82,81,83,93],[83,71,82,84,89],[84,83,85,90],[85,84,86,91,278],[86,85,8
7,91],[87,86,271,97],[88,70,271,116,131],[89,83,90,94],[90,84,89,91,95],[91,85,86,90,96],[92,272],[93,8
2,102,94],[94,89,93,95,103],[95,90,94,96],[96,91,95,97,107],[97,87,96,110],[98,78,99,279],[99,98,132,27
3],[100,81,101,134,273],[101,100],[102,93,103,274],[103,94,275,102],[104,274,105,118],[105,104,106,1
19],[106,105,275,120],[107,96,275,108,121],[108,107,109,122],[109,108,110,123],[110,97,109,111,124],
[111,112,110,125],[112,111,113,126],[113,112,114,127],[114,113,115,128,271],[115,114,116,129],[116,
88,115,130],[117,118,135,274],[118,104,117,137],[119,105,118,139],[120,106,119,141],[121,107,120,15
8],[122,108,121,160],[123,109,122,161],[124,110,123,162],[125,111,124,163],[126,112,125,164],[127,11
3,126,165],[128,114,127,129,166],[129,115,128,130,167],[130,116,129,131,142],[131,88,130,143],[132,
99,133,152],[133,273,132,134,153],[134,100,133,135],[135,117,134,136,154],[136,135,137,155],[137,13
6,118,138],[138,137,139,156],[139,119,138,140],[140,139,141,157],[141,120,140],[142,130,143,168],[14
3,131,142,169],[144,145],[145,144,146,187],[146,272,145,147,187],[147,76,146,148,276],[148,147,279,
189],[149,279,152,150,151],[150,151,149,192],[151,149,193],[152,149,132,153],[153,133,152,154,194],[
154,153,135,171,155],[155,136,154,156,172],[156,138,155,157,173],[157,156,158,140,174],[158,121,15
9],[159,160,175],[160,122,161],[161,123,162,176],[162,124,163,177],[163,125,164,178],[164,126,165,17
9],[165,127,166,180],[166,128,167,181],[167,165,168,129,182],[168,142,183],[169,143,170,207],[170,16
9,208],[171,154,172,197],[172,155,171,173],[173,156,172,174],[174,157,173,175,198],[175,159,174,176
],[176,161,175,177],[177,162,176,178,202],[178,163,177,179],[179,164,178,180],[180,165,179,181],[181
,166,180,182],[182,167,181,183,184],[183,168,182,185],[184,182,185,206],[185,183,184,186],[186,185],
[187,145,146,187,209,276],[188,189,276],[189,148,188,211],[190,191],[191,190,192,211],[192,150,191,
212],[193,151,194,213],[194,193,153,195,196],[195,194,214,215],[196,194,197,216],[197,171,196,198,2
17],[198,174,197,199],[199,198,200,201],[200,199,201],[201,199,200,218],[202,177,203,204],[203,202,2
05,219],[204,202,205,206],[205,203,204,220],[206,184,204,252],[207,169,237,208,236],[208,170,207,25
5],[209,187,210,238],[210,276,209,211],[211,189,230],[212,191,192,213],[213,193,212,214,222],[214,19
```

5,213,215,231],[215,214,195,216],[216,196,215,217],[217,197,216,218],[218,201,217,219,224],[219,203,218,220,226],[220,205,219,221,227],[221,220],[222,213,223,230],[223,222],[224,218,225],[225,224,228,281],[226,219,227,228],[227,220,226,229],[228,225,226,229,232],[229,227,228,233],[230,211,222,223,231],[231,214,230,239],[232,228,233,235,281],[233,229,232,234],[234,233],[235,232,251,281],[236,207,237,280],[237,207,236],[238,209],[239,231,240,243],[240,239,241],[241,240,250],[242,243,246,247],[243,244,239,242,248],[244,243],[245,246],[246,245,242,247],[247,242,246,248],[248,243,247,249,263],[249,240,248,250],[250,241,249,251,256],[251,235,250,252,259],[252,206,251,255],[253,280,254,255],[254,280,253],[255,208,252,253],[256,250,257,264],[257,256],[258,259],[259,251,258,260],[260,259,277],[261,262],[262,261,263],[263,248,262,267],[264,256,263,265],[265,264,266,268],[266,259,265,277],[267,263],[268,265],[269,277],[270,62,69,70,271],[271,270,87,88,114],[272,92,74,146],[273,80,99,100,133],[274,102,104,117],[275,103,106,107],[276,147,187,188,210],[277,260,266,269],[278,69,71,85],[279,98,148,149],[280,236,253,254],[281,225,232,235]]

LaneList = [[1, 2, 0, 0], [1, 3, 1, 1], [2, 4, 1, 1], [2, 5, 2, 1], [3, 4, 1, 1], [3, 7, 1, 1], [4, 5, 1, 1], [4, 8, 1, 1], [6, 9, 1, 1], [7, 12, 1, 1], [7, 24, 1, 1], [8, 9, 1, 1], [8, 13, 2, 2], [9, 14, 1, 1], [10, 34, 1, 1], [11, 12, 1, 1], [11, 21, 1, 1], [12, 23, 1, 1], [13, 14, 1, 1], [14, 18, 1, 1], [15, 35, 1, 1], [16, 20, 1, 1], [17, 20, 0, 0], [17, 70, 1, 1], [18, 28, 1, 1], [18, 29, 1, 1], [19, 31, 1, 1], [19, 33, 1, 1], [20, 36, 0, 0], [21, 22, 2, 2], [22, 23, 2, 2], [22, 52, 1, 1], [23, 24, 2, 2], [23, 53, 1, 1], [24, 25, 2, 2], [24, 54, 2, 2], [25, 26, 2, 2], [25, 40, 2, 2], [26, 27, 1, 1], [26, 37, 0, 0], [27, 28, 2, 2], [27, 37, 1, 1], [28, 29, 2, 2], [29, 39, 1, 1], [29, 42, 0, 0], [30, 31, 1, 1], [30, 38, 1, 1], [31, 32, 1, 1], [32, 33, 1, 1], [32, 38, 1, 1], [33, 34, 1, 1], [34, 35, 1, 1], [35, 36, 1, 1], [36, 43, 0, 0], [37, 41, 0, 0], [38, 49, 0, 0], [39, 48, 1, 1], [39, 49, 1, 1], [40, 41, 1, 1], [40, 55, 1, 1], [41, 42, 1, 1], [41, 57, 2, 2], [43, 44, 0, 0], [43, 45, 1, 1], [44, 45, 1, 1], [44, 62, 0, 0], [46, 47, 1, 1], [46, 58, 1, 1], [47, 48, 1, 1], [47, 59, 1, 1], [48, 60, 1, 1], [49, 61, 0, 0], [50, 51, 1, 1], [50, 73, 2, 2], [51, 52, 2, 2], [51, 74, 1, 1], [52, 75, 1, 1], [53, 54, 1, 1], [53, 76, 1, 1], [54, 55, 2, 2], [55, 56, 2, 2], [55, 77, 1, 1], [56, 57, 1, 1], [56, 67, 1, 1], [57, 58, 2, 2], [57, 80, 0, 0], [58, 59, 3, 3], [58, 81, 1, 1], [59, 60, 3, 3], [60, 61, 3, 3], [61, 62, 2, 2], [61, 64, 0, 0], [62, 63, 1, 1], [62, 270, 0, 0], [64, 66, 0, 0], [64, 69, 1, 1], [65, 66, 1, 1], [66, 68, 0, 0], [67, 78, 1, 1], [67, 79, 1, 1], [68, 69, 1, 1], [68, 71, 2, 2], [69, 270, 1, 1], [69, 278, 1, 1], [70, 88, 1, 1], [70, 270, 1, 1], [71, 83, 0, 0], [71, 278, 1, 1], [72, 73, 2, 2], [73, 74, 1, 1], [74, 75, 2, 2], [74, 272, 1, 1], [75, 76, 2, 2], [76, 77, 2, 2], [76, 147, 1, 1], [77, 78, 2, 2], [78, 79, 2, 2], [79, 80, 3, 3], [80, 81, 2, 2], [80, 273, 0, 0], [81, 82, 2, 2], [81, 100, 2, 2], [82, 83, 2, 2], [82, 93, 2, 2], [83, 84, 1, 1], [83, 89, 0, 0], [84, 85, 1, 1], [84, 90, 1, 1], [85, 86, 1, 1], [85, 91, 1, 1], [85, 278, 1, 1], [86, 87, 1, 1], [86, 91, 1, 1], [87, 271, 1, 1], [87, 97, 1, 1], [88, 271, 1, 1], [88, 116, 1, 1], [88, 131, 1, 1], [89, 90, 1, 1], [89, 94, 0, 0], [90, 91, 0, 0], [90, 95, 1, 1], [91, 96, 1, 1], [92, 272, 1, 1], [93, 102, 1, 1], [93, 94, 1, 1], [94, 95, 0, 0], [94, 103, 0, 0], [95, 96, 0, 0], [96, 97, 0, 0], [96, 107, 0, 0], [97, 110, 1, 1], [98, 99, 1, 1], [98, 279, 0, 0], [99, 132, 2, 2], [99, 273, 1, 1], [100, 101, 1, 1], [100, 134, 1, 1], [100, 273, 1, 1], [102, 103, 1, 1], [102, 274, 1, 1], [103, 275, 0, 0], [104, 274, 1, 1], [104, 105, 1, 1], [104, 118, 1, 1], [105, 106, 1, 1], [105, 119, 1, 1], [106, 275, 1, 1], [106, 120, 1, 1], [107, 275, 1, 1], [107, 108, 1, 1], [107, 121, 1, 1], [108, 109, 1, 1], [108, 122, 1, 1], [109, 110, 1, 1], [109, 123, 1, 1], [110, 111, 1, 1], [110, 124, 1, 1], [111, 112, 1, 1], [111, 125, 1, 1], [112, 113, 1, 1], [112, 126, 1, 1], [113, 114, 1, 1], [113, 127, 1, 1], [114, 115, 1, 1], [114, 128, 1, 1], [114, 271, 1, 1], [115, 116, 1, 1], [115, 129, 1, 1], [116, 130, 1, 1], [117, 118, 0, 2], [117, 135, 3, 1], [117, 274, 1, 1], [118, 119, 0, 2], [118, 137, 1, 1], [119, 120, 0, 2], [119, 139, 1, 1], [120, 121, 0, 2], [120, 141, 1, 1], [121, 122, 0, 2], [121, 158, 1, 1], [122, 123, 0, 2], [122, 160, 1, 1], [123, 124, 0, 2], [123, 161, 1, 1], [124, 125, 0, 2], [124, 162, 1, 1], [125, 126, 0, 3], [125, 163, 1, 1], [126, 127, 0, 2], [126, 164, 1, 1], [127, 128, 0, 2], [127, 165, 1, 1], [128, 129, 1, 1], [128, 166, 2, 2], [129, 130, 1, 1], [129, 167, 1, 1], [130, 131,

1, 1], [130, 142, 1, 1], [131, 143, 1, 1], [132, 133, 0, 0], [132, 152, 0, 0], [133, 273, 0, 0], [133, 134, 0, 0], [133, 153, 0, 0], [134, 135, 0, 0], [135, 136, 0, 0], [135, 154, 2, 2], [136, 137, 0, 0], [136, 155, 1, 1], [137, 138, 0, 0], [138, 139, 0, 0], [138, 156, 1, 1], [139, 140, 0, 0], [140, 141, 0, 0], [140, 157, 2, 2], [142, 143, 0, 0], [142, 168, 1, 2], [143, 169, 0, 0], [144, 145, 2, 2], [145, 146, 2, 2], [145, 187, 1, 1], [146, 272, 3, 3], [146, 147, 1, 1], [146, 187, 4, 4], [147, 148, 2, 2], [147, 276, 1, 1], [148, 279, 2, 2], [148, 189, 1, 1], [149, 152, 0, 0], [149, 150, 0, 0], [149, 151, 2, 3], [149, 279, 2, 2], [150, 151, 1, 0], [150, 192, 0, 0], [151, 193, 2, 2], [152, 153, 1, 1], [153, 154, 1, 1], [153, 194, 3, 3], [154, 171, 1, 1], [154, 155, 1, 1], [155, 156, 1, 1], [155, 172, 1, 1], [156, 157, 1, 1], [156, 173, 1, 1], [157, 158, 2, 0], [157, 174, 1, 1], [158, 159, 2, 0], [159, 160, 2, 0], [159, 175, 1, 1], [160, 161, 2, 0], [161, 162, 2, 0], [161, 176, 1, 1], [162, 163, 2, 0], [162, 177, 1, 2], [163, 164, 2, 0], [163, 178, 1, 1], [164, 165, 2, 0], [164, 179, 1, 1], [165, 166, 2, 0], [165, 180, 1, 1], [166, 167, 2, 0], [166, 181, 1, 1], [167, 168, 2, 0], [167, 182, 1, 1], [168, 183, 1, 1], [169, 170, 0, 0], [169, 207, 2, 2], [170, 208, 1, 1], [171, 172, 1, 1], [171, 197, 1, 1], [172, 173, 1, 1], [173, 174, 1, 1], [174, 175, 1, 1], [174, 198, 1, 1], [175, 176, 1, 1], [176, 177, 1, 1], [177, 178, 1, 1], [177, 202, 1, 1], [178, 179, 1, 1], [179, 180, 1, 1], [180, 181, 1, 1], [181, 182, 1, 1], [182, 183, 1, 1], [182, 184, 1, 1], [183, 185, 1, 1], [184, 185, 1, 1], [184, 206, 1, 1], [185, 186, 1, 1], [187, 209, 2, 3], [187, 276, 1, 1], [188, 189, 1, 1], [188, 276, 1, 1], [189, 211, 3, 1], [190, 191, 2, 2], [191, 192, 0, 0], [191, 211, 0, 0], [192, 212, 1, 1], [193, 194, 1, 1], [193, 213, 1, 1], [194, 195, 3, 3], [194, 196, 1, 1], [195, 214, 3, 3], [195, 215, 1, 1], [196, 197, 1, 1], [196, 216, 1, 1], [197, 198, 1, 1], [197, 217, 1, 1], [198, 199, 1, 1], [199, 200, 1, 1], [199, 201, 1, 1], [200, 201, 1, 1], [201, 218, 1, 1], [202, 203, 1, 1], [202, 204, 1, 1], [203, 205, 1, 1], [203, 219, 1, 1], [204, 205, 1, 1], [204, 206, 1, 1], [205, 220, 1, 1], [206, 252, 1, 1], [207, 237, 1, 1], [207, 208, 1, 1], [207, 236, 2, 2], [208, 255, 1, 1], [209, 210, 0, 0], [209, 238, 2, 2], [210, 276, 1, 1], [210, 211, 0, 0], [211, 230, 2, 3], [212, 213, 1, 1], [213, 214, 3, 1], [213, 222, 1, 1], [214, 215, 1, 3], [214, 231, 3, 3], [215, 216, 1, 1], [216, 217, 1, 1], [217, 218, 1, 1], [218, 219, 1, 1], [218, 224, 1, 1], [219, 220, 1, 1], [219, 226, 1, 1], [220, 221, 1, 1], [220, 227, 1, 1], [222, 223, 1, 1], [222, 230, 1, 1], [224, 225, 1, 1], [225, 281, 1, 1], [225, 228, 1, 1], [226, 227, 1, 1], [226, 228, 1, 1], [227, 229, 1, 1], [228, 229, 1, 1], [228, 232, 1, 1], [229, 233, 1, 1], [230, 231, 2, 2], [231, 239, 3, 3], [232, 233, 1, 1], [232, 235, 1, 1], [232, 281, 1, 1], [233, 234, 1, 1], [235, 251, 1, 1], [235, 281, 1, 1], [236, 237, 1, 1], [236, 280, 2, 2], [239, 240, 1, 1], [239, 243, 3, 3], [240, 241, 1, 1], [241, 250, 2, 2], [242, 243, 1, 1], [242, 246, 1, 1], [242, 247, 1, 1], [243, 244, 1, 1], [243, 248, 3, 3], [245, 246, 1, 1], [246, 247, 1, 1], [247, 248, 3, 1], [248, 249, 1, 3], [248, 263, 2, 2], [249, 250, 2, 2], [250, 251, 2, 2], [250, 256, 1, 1], [251, 252, 2, 2], [251, 259, 1, 3], [252, 255, 2, 2], [253, 254, 1, 1], [253, 255, 1, 1], [256, 257, 1, 1], [256, 264, 1, 1], [258, 259, 1, 1], [259, 260, 1, 1], [260, 277, 1, 1], [261, 262, 2, 2], [262, 263, 2, 2], [263, 267, 2, 1], [264, 265, 1, 1], [265, 266, 1, 1], [265, 268, 1, 1], [266, 277, 1, 1], [269, 277, 1, 1], [270, 271, 1, 1], [280, 253, 1, 1], [280, 254, 1, 1]]

CostList = [[1, 2, 871], [1, 3, 1130], [2, 4, 599], [2, 5, 701], [3, 4, 667], [3, 7, 808], [4, 5, 420], [4, 8, 1360], [6, 9, 357], [7, 12, 1100], [7, 24, 805], [8, 9, 284], [8, 13, 362], [9, 14, 366], [10, 34, 1160], [11, 12, 805], [11, 21, 556], [12, 23, 500], [13, 14, 280], [14, 18, 217], [15, 35, 331], [16, 20, 300], [17, 20, 606], [17, 70, 1960], [18, 28, 256], [18, 29, 351], [19, 31, 201], [19, 33, 310], [20, 36, 250], [21, 22, 402], [22, 23, 403], [22, 52, 800], [23, 24, 810], [23, 53, 817], [24, 25, 800], [24, 54, 796], [25, 26, 553], [25, 40, 398], [26, 27, 131], [26, 37, 230], [27, 28, 130], [27, 37, 226], [28, 29, 174], [29, 39, 751], [29, 42, 392], [30, 31, 528], [30, 38, 430], [31, 32, 76], [32, 33, 34], [32, 38, 542], [33, 34, 387], [34, 35, 301], [35, 36, 200], [36, 43, 440], [37, 41, 210], [38, 49, 515], [39, 48, 271], [39, 49, 553], [40, 41, 763], [40, 55, 410], [41, 42, 269], [41, 57, 392], [43, 44, 173], [43, 45, 345], [44, 45, 214], [44, 62, 372], [46, 47, 142], [46, 58, 247], [47, 48, 277], [47, 59, 236], [48, 60, 234], [49, 61, 148], [50, 51, 801], [50, 73, 785],

[51, 52, 420], [51, 74, 783], [52, 75, 808], [53, 54, 792], [53, 76, 800], [54, 55, 787], [55, 56, 157], [55, 77, 812], [56, 57, 654], [56, 67, 429], [57, 58, 428], [57, 80, 819], [58, 59, 148], [58, 81, 834], [59, 60, 283], [60, 61, 397], [61, 62, 1180], [61, 64, 277], [62, 63, 80], [62, 270, 327], [64, 66, 114], [64, 69, 391], [65, 66, 96], [66, 68, 110], [67, 78, 370], [67, 79, 672], [68, 69, 180], [68, 71, 113], [69, 270, 956], [69, 278, 108], [70, 88, 331], [70, 270, 249], [71, 83, 226], [71, 278, 177], [72, 73, 417], [73, 74, 793], [74, 75, 411], [74, 272, 197], [75, 76, 401], [76, 77, 1630], [76, 147, 813], [77, 78, 213], [78, 79, 339], [79, 80, 274], [80, 81, 406], [80, 273, 400], [81, 82, 422], [81, 100, 395], [82, 83, 397], [82, 93, 288], [83, 84, 95], [83, 89, 105], [84, 85, 84], [84, 90, 105], [85, 86, 93], [85, 91, 107], [85, 278, 233], [86, 87, 313], [86, 91, 199], [87, 271, 544], [87, 97, 278], [88, 271, 256], [88, 116, 466], [88, 131, 886], [89, 90, 93], [89, 94, 172], [90, 91, 181], [90, 95, 165], [91, 96, 169], [92, 272, 315], [93, 102, 111], [93, 94, 401], [94, 95, 95], [94, 103, 111], [95, 96, 88], [96, 97, 401], [96, 107, 191], [97, 110, 182], [98, 279, 431], [98, 99, 507], [99, 132, 371], [99, 273, 96], [100, 101, 544], [100, 134, 312], [100, 273, 407], [102, 103, 404], [102, 274, 80], [103, 275, 75], [104, 274, 81], [104, 105, 111], [104, 118, 110], [105, 106, 11], [105, 119, 109], [106, 275, 100], [106, 120, 107], [107, 275, 188], [107, 108, 194], [107, 121, 107], [108, 109, 101], [108, 122, 110], [109, 110, 99], [109, 123, 109], [110, 111, 116], [110, 124, 108], [111, 112, 130], [111, 125, 111], [112, 113, 110], [112, 126, 120], [113, 114, 182], [113, 127, 115], [114, 115, 131], [114, 128, 116], [114, 271, 462], [115, 116, 135], [115, 129, 118], [116, 130, 116], [117, 118, 82], [117, 135, 114], [117, 274, 114], [118, 119, 110], [118, 137, 115], [119, 120, 110], [119, 139, 115], [120, 121, 288], [120, 141, 115], [121, 122, 193], [121, 158, 114], [122, 123, 102], [122, 160, 114], [123, 124, 99], [123, 161, 114], [124, 125, 116], [124, 162, 121], [125, 126, 114], [125, 163, 121], [126, 127, 110], [126, 164, 59], [127, 128, 181], [127, 165, 123], [128, 129, 133], [128, 166, 122], [129, 130, 134], [129, 167, 122], [130, 131, 229], [130, 142, 123], [131, 143, 106], [132, 133, 184], [132, 152, 196], [133, 273, 322], [133, 134, 425], [133, 153, 94], [134, 135, 427], [135, 136, 69], [135, 154, 109], [136, 137, 27], [136, 155, 108], [137, 138, 85], [138, 139, 25], [138, 156, 106], [139, 140, 77], [140, 141, 32], [140, 157, 150], [142, 143, 195], [142, 168, 108], [143, 169, 606], [144, 145, 988], [145, 146, 171], [145, 187, 522], [146, 272, 1230], [146, 147, 1630], [146, 187, 395], [147, 148, 810], [147, 276, 590], [148, 279, 956], [148, 189, 682], [149, 152, 99], [149, 150, 72], [149, 151, 119], [149, 279, 320], [150, 151, 84], [150, 192, 426], [151, 193, 299], [152, 153, 272], [153, 154, 839], [153, 194, 349], [154, 171, 118], [154, 155, 84], [155, 156, 110], [155, 172, 55], [156, 157, 103], [156, 173, 53], [157, 158, 317], [157, 174, 117], [158, 159, 117], [159, 160, 77], [159, 175, 117], [160, 161, 99], [161, 162, 107], [161, 176, 110], [162, 163, 118], [162, 177, 110], [163, 164, 117], [163, 178, 108], [164, 165, 110], [164, 179, 107], [165, 166, 176], [165, 180, 108], [166, 167, 137], [166, 181, 112], [167, 168, 135], [167, 182, 112], [168, 183, 110], [169, 170, 169], [169, 207, 777], [170, 208, 883], [171, 172, 92], [171, 197, 225], [172, 173, 107], [173, 174, 102], [174, 175, 500], [174, 198, 222], [175, 176, 107], [176, 177, 107], [177, 178, 121], [177, 202, 328], [178, 179, 112], [179, 180, 109], [180, 181, 173], [181, 182, 138], [182, 183, 138], [182, 184, 108], [183, 185, 109], [184, 185, 141], [184, 206, 273], [185, 186, 110], [187, 209, 407], [187, 276, 880], [188, 189, 307], [188, 276, 804], [189, 211, 122], [190, 191, 77], [191, 192, 541], [191, 211, 547], [192, 212, 181], [193, 194, 400], [193, 213, 381], [194, 195, 217], [194, 196, 426], [195, 214, 252], [195, 215, 495], [196, 197, 408], [196, 216, 462], [197, 198, 304], [197, 217, 467], [198, 199, 202], [199, 200, 210], [199, 201, 111], [200, 201, 102], [201, 218, 127], [202, 203, 109], [202, 204, 160], [203, 205, 120], [203, 219, 233], [204, 205, 69], [204, 206, 537], [205, 220, 234], [206, 252, 1970], [207, 237, 337], [207, 208, 631], [207, 236, 723], [208, 255, 3690], [209, 210, 814], [209, 238, 834], [210, 276, 199], [210, 211, 809], [211, 230, 1680], [212, 213, 266], [213, 214, 375], [213, 222, 339], [214, 215, 323], [214, 231, 392], [215, 216, 91], [216,

217, 408], [217, 218, 309], [218, 219, 708], [218, 224, 226], [219, 220, 117], [219, 226, 120], [220, 221, 134], [220, 227, 121], [222, 223, 109], [222, 230, 114], [224, 225, 201], [225, 231, 107], [225, 228, 505], [226, 227, 122], [226, 228, 109], [227, 229, 109], [228, 229, 120], [228, 232, 222], [229, 233, 224], [230, 231, 276], [231, 239, 403], [232, 233, 95], [232, 235, 220], [233, 234, 399], [235, 251, 581], [236, 237, 626], [236, 242, 369], [239, 240, 420], [239, 243, 203], [240, 241, 412], [241, 250, 412], [242, 243, 111], [242, 246, 340], [242, 247, 217], [243, 244, 52], [243, 248, 221], [245, 246, 130], [246, 247, 125], [247, 248, 110], [248, 249, 156], [248, 263, 1200], [249, 250, 465], [250, 251, 841], [250, 256, 532], [251, 252, 1330], [251, 259, 858], [252, 255, 360], [253, 254, 624], [253, 255, 91], [256, 257, 229], [256, 264, 660], [258, 259, 547], [259, 260, 169], [260, 277, 276], [261, 262, 278], [262, 263, 500], [263, 267, 817], [264, 265, 491], [265, 266, 402], [265, 268, 1610], [266, 277, 156], [269, 277, 1210], [270, 271, 332], [280, 253, 737], [280, 254, 681], [281, 225, 100], [281, 232, 100], [281, 235, 100]]
TurnList = [[1, 2, 4], [4, 2, 1], [1, 3, 4], [4, 3, 1], [2, 1, 3], [3, 1, 2], [2, 4, 3], [3, 4, 2], [2, 4, 5], [5, 4, 2], [3, 4, 8], [8, 4, 3], [3, 7, 12], [12, 7, 3], [4, 8, 13], [13, 8, 4], [5, 4, 8], [8, 4, 5], [6, 9, 8], [8, 9, 6], [7, 12, 11], [11, 12, 7], [7, 24, 23], [23, 24, 7], [7, 24, 25], [25, 24, 7], [8, 13, 14], [14, 13, 8], [8, 9, 14], [14, 9, 8], [9, 8, 13], [13, 8, 9], [9, 14, 13], [13, 14, 9], [10, 34, 33], [33, 34, 10], [10, 34, 35], [35, 34, 10], [11, 21, 22], [22, 21, 11], [11, 12, 23], [23, 12, 11], [12, 23, 22], [22, 23, 12], [12, 23, 24], [24, 23, 12], [12, 7, 24], [24, 7, 12], [13, 14, 18], [18, 14, 13], [14, 18, 29], [29, 18, 14], [15, 35, 34], [34, 35, 15], [15, 35, 36], [36, 35, 15], [16, 20, 17], [17, 20, 16], [16, 20, 36], [36, 20, 16], [17, 70, 270], [270, 70, 17], [18, 29, 28], [28, 29, 18], [18, 28, 27], [27, 28, 18], [19, 31, 30], [30, 31, 19], [19, 31, 32], [32, 31, 19], [19, 33, 32], [32, 33, 19], [19, 33, 34], [34, 33, 19], [20, 17, 70], [70, 17, 20], [20, 36, 35], [35, 36, 20], [21, 22, 52], [52, 22, 21], [22, 23, 53], [53, 23, 22], [22, 52, 51], [51, 52, 22], [23, 24, 54], [54, 24, 23], [24, 23, 53], [53, 23, 24], [24, 25, 40], [40, 25, 24], [24, 54, 53], [53, 54, 24], [25, 24, 54], [54, 24, 25], [25, 40, 41], [41, 40, 25], [25, 26, 37], [37, 26, 25], [26, 27, 37], [37, 27, 26], [26, 37, 27], [27, 37, 26], [29, 42, 41], [41, 42, 29], [29, 39, 48], [48, 39, 29], [30, 38, 32], [32, 38, 30], [31, 32, 38], [38, 32, 31], [31, 30, 38], [38, 30, 31], [31, 19, 33], [33, 19, 31], [35, 36, 43], [43, 36, 35], [36, 43, 45], [45, 43, 36], [37, 41, 40], [40, 41, 37], [37, 41, 42], [42, 41, 37], [38, 49, 39], [39, 49, 38], [39, 48, 47], [47, 48, 39], [39, 49, 61], [61, 49, 39], [40, 55, 54], [54, 55, 40], [40, 55, 56], [56, 55, 40], [40, 41, 57], [57, 41, 40], [41, 57, 56], [56, 57, 41], [41, 57, 58], [58, 57, 41], [42, 41, 57], [57, 41, 42], [43, 44, 45], [45, 44, 43], [43, 45, 44], [44, 45, 43], [45, 44, 62], [62, 44, 45], [44, 43, 45], [45, 43, 44], [44, 62, 63], [63, 62, 44], [44, 62, 61], [61, 62, 44], [46, 47, 59], [59, 47, 46], [46, 58, 57], [57, 58, 46], [46, 58, 59], [59, 58, 46], [47, 46, 58], [58, 46, 47], [47, 48, 60], [60, 48, 47], [47, 59, 58], [58, 59, 47], [47, 59, 60], [60, 59, 47], [48, 47, 59], [59, 47, 48], [48, 39, 49], [49, 39, 48], [48, 60, 59], [59, 60, 48], [48, 60, 61], [61, 60, 48], [49, 61, 60], [60, 61, 49], [49, 61, 62], [62, 61, 49], [50, 51, 74], [74, 51, 50], [50, 73, 72], [72, 73, 50], [50, 73, 74], [74, 73, 50], [51, 52, 75], [75, 52, 51], [51, 74, 73], [73, 74, 51], [51, 74, 75], [75, 74, 51], [51, 50, 73], [73, 50, 51], [52, 75, 76], [76, 75, 52], [52, 75, 74], [74, 75, 52], [52, 51, 74], [74, 51, 52], [54, 53, 76], [76, 53, 54], [54, 55, 77], [77, 55, 54], [56, 55, 77], [77, 55, 56], [56, 67, 79], [79, 67, 56], [56, 57, 80], [80, 57, 56], [57, 56, 67], [67, 56, 57], [57, 58, 81], [81, 58, 57], [57, 80, 79], [79, 80, 57], [57, 80, 81], [81, 80, 57], [58, 57, 80], [80, 57, 58], [58, 81, 80], [80, 81, 58], [58, 81, 82], [82, 81, 58], [59, 58, 81], [81, 58, 59], [60, 61, 64], [64, 61, 60], [61, 62, 270], [270, 62, 61], [61, 64, 69], [69, 64, 61], [62, 61, 64], [64, 61, 62], [62, 270, 69], [69, 270, 62], [62, 270, 70], [70, 270, 62], [63, 62, 270], [270, 62, 63], [64, 66, 65], [65, 66, 64], [64, 69, 68], [68, 69, 64], [64, 69, 270], [270, 69, 64], [65, 66, 68], [68, 66, 65], [66, 64, 69], [69, 64, 66], [66, 68, 69], [69, 68, 66], [67, 78, 77], [77, 78, 67], [67, 78, 79], [79, 78, 67], [67, 79, 78], [78, 79, 67], [67, 79, 80], [80, 79, 67], [68, 71, 278], [278, 71, 68], [68, 69, 278], [278, 69, 68], [69,

68, 71], [71, 68, 69], [69, 278, 71], [71, 278, 69], [70, 270, 271], [271, 270, 70], [70, 88, 271], [271, 88, 70], [70, 88, 131], [131, 88, 70], [71, 278, 85], [85, 278, 71], [71, 83, 82], [82, 83, 71], [71, 83, 84], [84, 83, 71], [73, 74, 272], [272, 74, 73], [74, 272, 92], [92, 272, 74], [75, 74, 272], [272, 74, 75], [75, 76, 147], [147, 76, 75], [76, 147, 146], [146, 147, 76], [76, 147, 148], [148, 147, 76], [77, 76, 147], [147, 76, 77], [78, 67, 79], [79, 67, 78], [79, 80, 273], [273, 80, 79], [80, 273, 99], [99, 273, 80], [80, 273, 100], [100, 273, 80], [80, 81, 100], [100, 81, 80], [81, 100, 273], [273, 100, 81], [81, 100, 101], [101, 100, 81], [81, 82, 93], [93, 82, 81], [82, 81, 100], [100, 81, 82], [82, 83, 89], [89, 83, 82], [82, 93, 94], [94, 93, 82], [83, 82, 93], [93, 82, 83], [83, 84, 90], [90, 84, 83], [83, 89, 90], [90, 89, 83], [84, 83, 89], [89, 83, 84], [84, 85, 91], [91, 85, 84], [84, 90, 89], [89, 90, 84], [84, 90, 91], [91, 90, 84], [85, 86, 91], [91, 86, 85], [85, 84, 90], [90, 84, 85], [85, 91, 86], [86, 91, 85], [85, 91, 90], [90, 91, 85], [86, 87, 97], [97, 87, 86], [86, 85, 278], [278, 85, 86], [86, 85, 91], [91, 85, 86], [87, 86, 91], [91, 86, 87], [87, 271, 270], [270, 271, 87], [87, 271, 114], [114, 271, 87], [88, 271, 270], [270, 271, 88], [88, 70, 270], [270, 70, 88], [88, 131, 130], [130, 131, 88], [88, 116, 115], [115, 116, 88], [89, 94, 93], [93, 94, 89], [89, 94, 95], [95, 94, 89], [89, 90, 95], [95, 90, 89], [90, 89, 94], [94, 89, 90], [90, 95, 94], [94, 95, 90], [90, 95, 96], [96, 95, 90], [90, 91, 96], [96, 91, 90], [91, 90, 95], [95, 90, 91], [91, 96, 97], [97, 96, 91], [91, 96, 95], [95, 96, 91], [92, 272, 146], [146, 272, 92], [93, 94, 103], [103, 94, 93], [93, 102, 103], [103, 102, 93], [94, 93, 102], [102, 93, 94], [94, 103, 102], [102, 103, 94], [95, 96, 107], [107, 96, 95], [96, 97, 110], [110, 97, 96], [97, 96, 107], [107, 96, 97], [97, 110, 109], [109, 110, 97], [97, 110, 111], [111, 110, 97], [98, 99, 132], [132, 99, 98], [99, 132, 152], [152, 132, 99], [99, 132, 133], [133, 132, 99], [99, 273, 133], [133, 273, 99], [100, 273, 133], [133, 273, 100], [100, 134, 133], [133, 134, 100], [100, 134, 135], [135, 134, 100], [101, 100, 134], [134, 100, 101], [102, 103, 275], [275, 103, 102], [102, 274, 104], [104, 274, 102], [103, 102, 274], [274, 102, 103], [103, 275, 106], [106, 275, 103], [103, 275, 107], [107, 275, 103], [104, 274, 117], [117, 274, 104], [104, 118, 117], [117, 118, 104], [104, 105, 119], [119, 105, 104], [105, 104, 118], [118, 104, 105], [105, 106, 120], [120, 106, 105], [106, 105, 119], [119, 105, 106], [107, 108, 122], [122, 108, 107], [108, 107, 121], [121, 107, 108], [108, 109, 123], [123, 109, 108], [109, 108, 122], [122, 108, 109], [110, 109, 123], [123, 109, 110], [110, 111, 125], [125, 111, 110], [111, 110, 124], [124, 110, 111], [111, 112, 126], [126, 112, 111], [112, 111, 125], [125, 111, 112], [112, 113, 127], [127, 113, 112], [113, 112, 126], [126, 112, 113], [113, 114, 128], [128, 114, 113], [114, 113, 127], [127, 113, 114], [114, 128, 129], [129, 128, 114], [114, 115, 129], [129, 115, 114], [115, 114, 128], [128, 114, 115], [115, 129, 128], [128, 129, 115], [115, 129, 130], [130, 129, 115], [115, 116, 130], [130, 116, 115], [116, 115, 129], [129, 115, 116], [116, 130, 129], [129, 130, 116], [116, 130, 131], [131, 130, 116], [117, 135, 136], [136, 135, 117], [118, 137, 136], [136, 137, 118], [118, 117, 135], [135, 117, 118], [118, 137, 138], [138, 137, 118], [119, 139, 138], [138, 139, 119], [119, 139, 140], [140, 139, 119], [120, 141, 140], [140, 141, 120], [128, 129, 167], [167, 129, 128], [129, 128, 166], [166, 128, 129], [129, 130, 142], [142, 130, 129], [130, 129, 167], [167, 129, 130], [130, 142, 143], [143, 142, 130], [130, 131, 143], [143, 131, 130], [131, 130, 142], [142, 130, 131], [131, 143, 169], [169, 143, 131], [131, 143, 142], [142, 143, 131], [132, 99, 273], [273, 99, 132], [132, 133, 273], [273, 133, 132], [132, 133, 153], [153, 133, 132], [132, 152, 153], [153, 152, 132], [133, 153, 154], [154, 153, 133], [134, 133, 153], [153, 133, 134], [134, 135, 154], [154, 135, 134], [135, 136, 155], [155, 136, 135], [135, 154, 153], [153, 154, 135], [135, 154, 155], [155, 154, 135], [136, 155, 154], [154, 155, 136], [136, 135, 154], [154, 135, 136], [136, 155, 156], [156, 155, 136], [137, 136, 155], [155, 136, 137], [137, 138, 156], [156, 138, 137], [138, 156, 155], [155, 156, 138], [138, 156, 157], [157, 156, 138], [139, 138, 156], [156, 138, 139], [139, 140, 157], [157, 140, 139], [140, 157, 156], [156, 157, 140], [141, 140, 157], [157, 140, 141], [143, 142, 168], [168, 142, 143], [143, 169, 207], [207, 169, 143], [144,

145, 187], [187, 145, 144], [145, 146, 187], [187, 146, 145], [145, 187, 146], [146, 187, 145], [145, 187, 209], [209, 187, 145], [146, 145, 187], [187, 145, 146], [146, 187, 276], [276, 187, 146], [146, 147, 276], [276, 147, 146], [147, 276, 187], [187, 276, 147], [147, 148, 189], [189, 148, 147], [147, 276, 188], [188, 276, 147], [148, 189, 188], [188, 189, 148], [148, 147, 276], [276, 147, 148], [149, 152, 153], [153, 152, 149], [150, 192, 212], [212, 192, 150], [150, 149, 279], [279, 149, 150], [150, 149, 151], [151, 149, 150], [151, 149, 152], [152, 149, 151], [151, 193, 194], [194, 193, 151], [152, 153, 194], [194, 153, 152], [152, 149, 279], [279, 149, 152], [153, 194, 193], [193, 194, 153], [153, 194, 196], [196, 194, 153], [153, 154, 171], [171, 154, 153], [154, 153, 194], [194, 153, 154], [154, 171, 172], [172, 171, 154], [154, 155, 172], [172, 155, 154], [155, 154, 171], [171, 154, 155], [155, 172, 171], [171, 172, 155], [155, 172, 173], [173, 172, 155], [155, 156, 173], [173, 156, 155], [156, 155, 172], [172, 155, 156], [156, 173, 172], [172, 173, 156], [156, 173, 174], [174, 173, 156], [156, 157, 174], [174, 157, 156], [157, 174, 173], [173, 174, 157], [157, 156, 173], [173, 156, 157], [157, 174, 175], [175, 174, 157], [159, 175, 176], [176, 175, 159], [161, 176, 175], [175, 176, 161], [161, 176, 177], [177, 176, 161], [162, 177, 176], [176, 177, 162], [162, 177, 178], [178, 177, 162], [163, 178, 177], [177, 178, 163], [163, 178, 179], [179, 178, 163], [164, 179, 178], [178, 179, 164], [164, 179, 180], [180, 179, 164], [165, 180, 179], [179, 180, 165], [165, 180, 181], [181, 180, 165], [166, 181, 180], [180, 181, 166], [166, 181, 182], [182, 181, 166], [167, 182, 181], [181, 182, 167], [167, 182, 183], [183, 182, 167], [168, 183, 182], [182, 183, 168], [169, 170, 208], [208, 170, 169], [169, 207, 237], [237, 207, 169], [169, 207, 208], [208, 207, 169], [170, 169, 207], [207, 169, 170], [170, 208, 207], [207, 208, 170], [171, 197, 196], [196, 197, 171], [171, 197, 198], [198, 197, 171], [173, 174, 198], [198, 174, 173], [174, 198, 197], [197, 198, 174], [175, 174, 198], [198, 174, 175], [176, 177, 202], [202, 177, 176], [177, 202, 204], [204, 202, 177], [178, 177, 202], [202, 177, 178], [181, 182, 184], [184, 182, 181], [182, 183, 185], [185, 183, 182], [182, 184, 185], [185, 184, 182], [183, 182, 184], [184, 182, 183], [183, 185, 184], [184, 185, 183], [184, 185, 186], [186, 185, 184], [184, 206, 204], [204, 206, 184], [185, 184, 206], [206, 184, 185], [187, 209, 210], [210, 209, 187], [187, 276, 210], [210, 276, 187], [188, 276, 210], [210, 276, 188], [188, 189, 211], [211, 189, 188], [189, 188, 276], [276, 188, 189], [189, 148, 279], [279, 148, 189], [190, 191, 192], [192, 191, 190], [191, 192, 212], [212, 192, 191], [192, 212, 213], [213, 212, 192], [193, 194, 195], [195, 194, 193], [193, 213, 212], [212, 213, 193], [193, 213, 214], [214, 213, 193], [195, 214, 213], [213, 214, 195], [195, 214, 215], [215, 214, 195], [195, 194, 196], [196, 194, 195], [196, 216, 215], [215, 216, 196], [196, 216, 217], [217, 216, 196], [196, 197, 217], [217, 197, 196], [197, 196, 216], [216, 196, 197], [197, 217, 216], [216, 217, 197], [197, 217, 218], [218, 217, 197], [197, 198, 199], [199, 198, 197], [198, 197, 217], [217, 197, 198], [198, 199, 200], [200, 199, 198], [199, 200, 201], [201, 200, 199], [200, 201, 218], [218, 201, 200], [201, 218, 217], [217, 218, 201], [201, 218, 219], [219, 218, 201], [202, 204, 205], [205, 204, 202], [202, 204, 206], [206, 204, 202], [202, 203, 205], [205, 203, 202], [203, 202, 204], [204, 202, 203], [203, 205, 204], [204, 205, 203], [203, 219, 218], [218, 219, 203], [203, 219, 220], [220, 219, 203], [203, 205, 220], [220, 205, 203], [204, 206, 252], [252, 206, 204], [205, 204, 206], [206, 204, 205], [205, 203, 219], [219, 203, 205], [205, 220, 219], [219, 220, 205], [205, 220, 221], [221, 220, 205], [206, 252, 251], [251, 252, 206], [206, 252, 255], [255, 252, 206], [207, 237, 236], [236, 237, 207], [207, 236, 237], [237, 236, 207], [207, 208, 255], [255, 208, 207], [208, 207, 236], [236, 207, 208], [208, 255, 253], [253, 255, 208], [209, 187, 276], [276, 187, 209], [209, 210, 276], [276, 210, 209], [210, 209, 238], [238, 209, 210], [211, 230, 222], [222, 230, 211], [212, 213, 222], [222, 213, 212], [213, 222, 223], [223, 222, 213], [213, 214, 231], [231, 214, 213], [214, 213, 222], [222, 213, 214], [214, 231, 230], [230, 231, 214], [214, 195, 215], [215, 195, 214], [217, 218, 224], [224, 218, 217], [218, 224, 225], [225, 224, 218], [218, 219, 226], [226, 219, 218], [219, 218, 224], [224, 218, 219], [219, 226,

```
227], [227, 226, 219], [219, 220, 227], [227, 220, 219], [220, 227, 226], [226, 227, 220], [220, 219, 226],
[226, 219, 220], [221, 220, 227], [227, 220, 221], [222, 230, 231], [231, 230, 222], [223, 222, 230], [230,
222, 223], [224, 225, 281], [281, 225, 224], [225, 281, 232], [232, 281, 225], [225, 228, 232], [232, 228,
225], [226, 227, 229], [229, 227, 226], [226, 228, 229], [229, 228, 226], [227, 229, 228], [228, 229, 227],
[228, 229, 233], [233, 229, 228], [228, 232, 233], [233, 232, 228], [228, 225, 281], [281, 225, 228], [228,
232, 281], [281, 232, 228], [229, 228, 232], [232, 228, 229], [229, 233, 232], [232, 233, 229], [229, 233,
234], [234, 233, 229], [230, 231, 239], [239, 231, 230], [231, 239, 240], [240, 239, 231], [232, 281, 235],
[235, 281, 232], [233, 232, 235], [235, 232, 233], [235, 251, 250], [250, 251, 235], [235, 251, 252], [252,
251, 235], [236, 280, 254], [254, 280, 236], [236, 207, 237], [237, 207, 236], [237, 236, 280], [280, 236,
237], [239, 243, 244], [244, 243, 239], [239, 243, 242], [242, 243, 239], [240, 239, 243], [243, 239, 240],
[240, 241, 250], [250, 241, 240], [242, 246, 245], [245, 246, 242], [242, 246, 247], [247, 246, 242], [242,
247, 248], [248, 247, 242], [242, 243, 248], [248, 243, 242], [243, 248, 247], [247, 248, 243], [243, 248,
249], [249, 248, 243], [244, 243, 248], [248, 243, 244], [246, 242, 247], [247, 242, 246], [247, 248, 263],
[263, 248, 247], [248, 263, 262], [262, 263, 248], [249, 250, 256], [256, 250, 249], [250, 251, 259], [259,
251, 250], [250, 256, 257], [257, 256, 250], [251, 259, 260], [260, 259, 251], [251, 259, 258], [258, 259,
251], [252, 251, 259], [259, 251, 252], [252, 255, 253], [253, 255, 252], [253, 254, 280], [280, 254, 253],
[253, 280, 254], [254, 280, 253], [254, 253, 255], [255, 253, 254], [256, 264, 265], [265, 264, 256], [257,
256, 264], [264, 256, 257], [260, 277, 266], [266, 277, 260], [260, 277, 269], [269, 277, 260], [262, 263,
267], [267, 263, 262], [264, 265, 268], [268, 265, 264], [266, 277, 269], [269, 277, 266], [266, 265, 268],
[268, 265, 266]]
kInt = 6
lInt = 8

#------------------------------------------------------------------------------------------------------------
#Function definitions
#------------------------------------------------------------------------------------------------------------

def CreateCostMatrix(NodeList, CostList):
    ReturnList = []
    for n in range(len(NodeList)):
        Row = []
        for m in range(len(NodeList)):
            Row.append(999999)
        ReturnList.append(Row)
    for Pair in CostList:
        i = (Pair[0] - 1)
        j = (Pair[1] - 1)
        c = Pair[2]
        ReturnList[i][j] = c
        ReturnList[j][i] = c
    return ReturnList

def CreateLaneMatrix(NodeList, LaneList):
```

```
    ReturnList = []
    for n in range(len(NodeList)):
        Row = []
        for m in range(len(NodeList)):
            Row.append(0)
        ReturnList.append(Row)
    for Pair in LaneList:
        i = (Pair[0] - 1)
        j = (Pair[1] - 1)
        To = Pair[2]
        From = Pair[3]
        ReturnList[i][j] = To
        ReturnList[j][i] = From
    return ReturnList

def WriteOutputFile(List, File):            #Writes the contents of a list to a file
    OutputFile = open(File, "w")            #Opens the file
    for Item in List:           #Loops through the list
        OutputFile.write(str(Item) + "\n")          #Writes each item to the file
    OutputFile.close()          #Close the file
    return True         #Returns true to indicate success

def CheckListFor(List, Item):
    IsThere = False
    for Thing in List:
        if Thing == Item:
            IsThere = True
    return IsThere

def MakeT(i, j):
    ReturnString = "T[" + str(i) + ", " + str(j) + ", k, l]"
    return ReturnString

#-------------------------------------------------------------------------------------------------------------------
#Main
#-------------------------------------------------------------------------------------------------------------------

PairList = []

for Node in NodeList:
    for j in Node:
        if j != Node[0]:
            Pair = [Node[0], j]
```

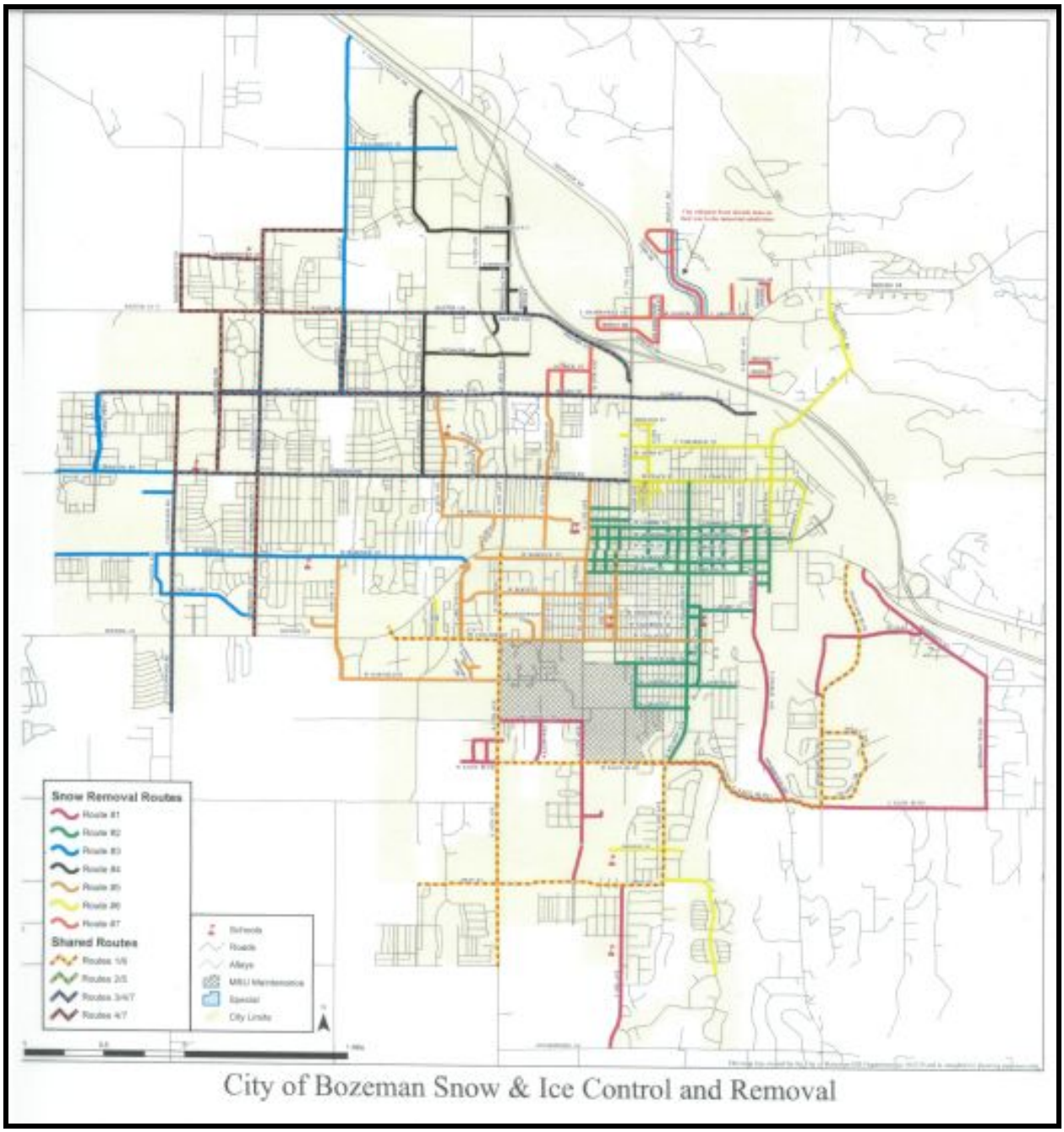
```
            if CheckListFor(PairList, Pair) == False:
                PairList.append(Pair)

print(PairList)

MyString = "{"

First = True

for Pair in PairList:
    MyString += "(" + str(Pair[0]) + ", " + str(Pair[1]) + "), "

MyString += "}"

print(MyString)

TotalCost = 0

for Arc in LaneList:
    if Arc[2] > 0:
        Pair = [Arc[0], Arc[1]]
        if CheckListFor(PairList, Pair) == False:
            print(Pair)
    if Arc[3] > 0:
        Pair = [Arc[1], Arc[0]]
        if CheckListFor(PairList, Pair) == False:
            print(Pair)

for Shit in CostList:
    for InnerShit in Shit:
        TotalCost += int(InnerShit)

print(TotalCost)

MyNodeList = []

for Node in NodeList:
    MyNodeList.append(Node[0])

'''

MyTurnList = []
```

```
for Turn in TurnList:
    SFY = False
    MyArc = [Turn[0], Turn[1]]
    if CheckListFor(PairList, MyArc) == False:
        print(MyArc)
        SFY = True
    MyArc = [Turn[1], Turn[0]]
    if CheckListFor(PairList, MyArc) == False:
        print(MyArc)
        SFY = True
    MyArc = [Turn[1], Turn[2]]
    if CheckListFor(PairList, MyArc) == False:
        print(MyArc)
        SFY = True
    MyArc = [Turn[2], Turn[1]]
    if CheckListFor(PairList, MyArc) == False:
        print(MyArc)
        SFY = True
    if SFY == False:
        MyTurnList.append(Turn)

print(MyTurnList)

'''

TurnPenString = ""

for Turn in TurnList:
    ProductString = " + " + MakeT(Turn[0], Turn[1]) + " * " + MakeT(Turn[1], Turn[2])
    TurnPenString += ProductString

print(TurnPenString)

'''

FinalTurnList = []

for Turn in MyTurnList:
    ForwardTurn = [Turn[0], Turn[1], Turn[2]]
    ReverseTurn = [Turn[2], Turn[1], Turn[0]]
    if CheckListFor(FinalTurnList, ForwardTurn) == False:
        FinalTurnList.append(ForwardTurn)
    if CheckListFor(FinalTurnList, ReverseTurn) == False:
```

```
        FinalTurnList.append(ReverseTurn)
        #print("SFY")

print(FinalTurnList)

'''

'''

CostMatrix = CreateCostMatrix(NodeList, CostList)   #CostMatrix[i - 1][j-1] is cost from i to j
LaneMatrix = CreateLaneMatrix(NodeList, LaneList)
#print(CostMatrix)

#print(CostMatrix[7][12])
#print(CostMatrix[12][7])

ParamString = "param n := " + str(len(NodeList)) + ";\nparam o := " + str(len(NodeList)) + ";\nparam p :=
" + str(kInt) + ";\nparam q := " + str(lInt) + ";\n"

CString = "\nparam\tC"
LString = "\nparam\tL"

for n in range(len(NodeList)):
    CString += "\t" + str(n + 1)
    LString += "\t" + str(n + 1)

CString += ":="
LString +=  ":="

for i in range(len(NodeList)):
    CString += "\n"
    CString += str(i+1)
    LString += "\n"
    LString += str(i+1)
    for j in range(len(NodeList)):
        CString += "\t"
        CString += str(CostMatrix[i][j])
        LString += "\t"
        LString += str(LaneMatrix[i][j])

CString += ";\n"
LString += ";\n"
```

```
MyString = ParamString + CString + LString
MyList = [MyString]

WriteOutputFile(MyList, "DataFileWhot.txt")

print(ParamString)
print(CString)
print(LString)
#print(len(NodeList))

#TestList = [1, 1]

#print(len(TestList))

'''
for Node in NodeList:
    if Node[0] == NodeList.index(Node):
        print("SFY")
        print(NodeList.index(Node))
'''

m = input("Press enter to exit.")          #Used to prevent the console from closing before the user is ready

'''
```

Appendix I - Current State Paths



Appendix J - Iterations of the Model & Lessons Learned

Model Version	Solver	Submitted	Run Time	Termination	Corrective Action / Lesson Learned
45.4	CPLEX	5/2/2020 5:41	7:30:16	Reached set time limit	Feasible solution found
45.3	CPLEX	5/2/2020 5:41	7:30:17	Reached set time limit	Feasible solution found
45.2	CPLEX	5/2/2020 5:41	2:20:17	Program finished	Feasible solution found
45.1	CPLEX	5/2/2020 5:41	1:00:50	Program finished	Feasible solution found
45	CPLEX	5/2/2020 4:22	0:45:07	Program finished	Feasible solution found
44.5	CPLEX	5/2/2020 4:19	0:00:05	Syntax error	Altered syntax
45.4	CPLEX	5/2/2020 4:14	7:30:03	Reached set time limit	Values in .dat file changed
44.4	CPLEX	5/2/2020 4:14	0:00:02	Syntax error	Altered syntax
45.3	CPLEX	5/2/2020 4:09	7:30:03	Reached set time limit	Values in .dat file changed
44.3	CPLEX	5/2/2020 4:09	0:00:02	Syntax error	Altered syntax
45.2	CPLEX	5/2/2020 4:05	7:30:03	Reached set time limit	Values in .dat file changed

44.2	CPLEX	5/2/2020 4:05	0:00:02	Syntax error	Altered syntax
45.1	CPLEX	5/2/2020 3:56	7:30:07	Reached set time limit	Values in .dat file changed
44.1	CPLEX	5/2/2020 3:56	0:00:07	Syntax error	Altered syntax
45	CPLEX	5/2/2020 3:56	7:30:09	Reached set time limit	Values in .dat file changed
44	CPLEX	5/2/2020 3:56	0:00:09	Syntax error	Altered syntax
43.7	CPLEX	5/2/2020 2:35	7:30:13	Reached set time limit	Formulation still too large, relaxing equity constraint
43.6	CPLEX	5/2/2020 2:35	7:30:16	Reached set time limit	Partition still too large
43.5	CPLEX	5/2/2020 2:32	7:30:12	Reached set time limit	Partition still too large
43.4	CPLEX	5/2/2020 2:32	7:30:12	Reached set time limit	Partition still too large
43.3	CPLEX	5/1/2020 20:30	7:30:24	Reached set time limit	Partition still too large
43.2	CPLEX	5/1/2020 20:30	7:30:28	Reached set time limit	Partition still too large
43.1	CPLEX	5/1/2020 20:30	7:30:26	Reached set time limit	Partition still too large

43	CPLEX	5/1/2020 20:30	7:30:08	Reached set time limit	Partition still too large
42.5	CPLEX	5/1/2020 17:37	0:06:57	Syntax error	Altered syntax
42.4	CPLEX	5/1/2020 17:37	7:30:11	Reached set time limit	Partition still too large
42.3	CPLEX	5/1/2020 17:13	0:33:31	Syntax error	Altered syntax
42.2	CPLEX	5/1/2020 17:13	7:30:08	Reached set time limit	Partition still too large
42.1	CPLEX	5/1/2020 17:02	1:18:00	Done	Feasible solution found
42	CPLEX	5/1/2020 17:02	0:06:58	Syntax error	Altered syntax
41.7	CPLEX	5/1/2020 11:05	7:30:22	Reached set time limit	Reduced size of partitions and changed CPLEX settings
41.6	CPLEX	5/1/2020 11:05	7:30:22	Reached set time limit	Formulation still too large
41.5	CPLEX	5/1/2020 11:05	7:30:22	Reached set time limit	Formulation still too large
41.4	CPLEX	5/1/2020 11:05	7:30:13	Reached set time limit	Formulation still too large
41.3	CPLEX	5/1/2020 10:59	0:00:10	Syntax error	Altered syntax

41.2	CPLEX	5/1/2020 10:59	0:00:12	Syntax error	Altered syntax
41.1	CPLEX	5/1/2020 10:59	0:00:13	Syntax error	Altered syntax
41	CPLEX	5/1/2020 10:59	0:00:13	Syntax error	Altered syntax
40.4	CPLEX	5/1/2020 4:34	7:30:28	Reached set time limit	Formulation still too large, reduced size of partitions
40.3	CPLEX	5/1/2020 4:33	7:30:25	Reached set time limit	Formulation still too large
40.2	CPLEX	4/30/2020 21:31	7:30:12	Reached set time limit	Formulation still too large
40.1	CPLEX	4/30/2020 21:27	7:30:36	Reached set time limit	Formulation still too large
40	CPLEX	4/30/2020 21:27	7:30:32	Reached set time limit	Formulation still too large
39	CPLEX	4/30/2020 21:27	0:19:09	Program finished	New variables broke lane constraint, modified lane constraints
38.2	CPLEX	4/30/2020 15:24	7:00:42	Reached set time limit	Formulation still too large, changing variable definitions and CPLEX settings
38.1	CPLEX	4/30/2020 15:14	7:00:41	Reached set time limit	Formulation still too large
38	CPLEX	4/30/2020 15:13	0:00:02	Syntax error	Altered syntax

37.6	CPLEX	4/30/2020 15:13	5:33:31	Reached set time limit	Formulation still too large, changed variable definitions and CPLEX settings
37.5	CPLEX	4/30/2020 15:12	5:33:37	Reached set time limit	Formulation still too large
37.4	CPLEX	4/30/2020 15:12	5:33:48	Reached set time limit	Formulation still too large
37.3	CPLEX	4/30/2020 15:11	5:33:50	Reached set time limit	Formulation still too large
37.2	CPLEX	4/30/2020 15:11	7:00:22	Reached set time limit	Formulation still too large
37.1	CPLEX	4/30/2020 15:10	5:33:44	Reached set time limit	Formulation still too large
36.5	CPLEX	4/30/2020 1:02	8:00:16	Timed out	Formulation still too large, reducing size of partitions
36.4	CPLEX	4/30/2020 1:02	8:00:19	Timed out	Formulation still too large
36.3	CPLEX	4/30/2020 0:21	8:00:17	Timed out	Formulation still too large
36.2	CPLEX	4/30/2020 0:20	8:00:15	Timed out	Formulation still too large
36.1	CPLEX	4/30/2020 0:20	8:00:24	Timed out	Formulation still too large
36	CPLEX	4/30/2020 0:20	8:00:13	Timed out	Formulation still too large

35	CPLEX	4/30/2020 0:20	2:06:58	Program finished	Isolated loop problems arise again
34.6	CPLEX	4/30/2020 0:20	8:00:12	Timed out	Variable definitions changed
34.5	CPLEX	4/30/2020 0:20	8:00:20	Timed out	Values in .dat file changed
34.4	CPLEX	4/28/2020 23:39	8:00:31	Timed out	Values in .dat file changed
34.3	CPLEX	4/28/2020 23:38	8:00:56	Timed out	Values in .dat file changed
34.2	CPLEX	4/28/2020 23:38	8:01:16	Timed out	Values in .dat file changed
34.1	CPLEX	4/28/2020 23:37	8:00:15	Timed out	Values in .dat file changed
34	CPLEX	4/28/2020 23:37	8:00:18	Timed out	Values in .dat file changed
33.3	scip	4/28/2020 20:53	11:06:41	Timed out	Formulation still too large
31.8	CPLEX	4/28/2020 20:52	2:22:05	Program finished	Problem with isolated loops
31.7	CPLEX	4/28/2020 20:03	0:01:46	Syntax error	Altered syntax
31.6	CPLEX	4/28/2020 19:47	0:02:30	Syntax error	Altered syntax

33.2	scip	4/28/2020 19:45	5:17:34	Program finished	Problem with isolated loops found
33.1	scip	4/28/2020 19:44	0:00:12	Syntax error	Formulation still too large
31.5	CPLEX	4/28/2020 19:40	0:40:40	Program finished	Lane constraint not functioning as intended
33	scip	4/28/2020 18:57	0:00:19	Syntax error	Formulation still too large
32.1	CPLEX	4/28/2020 18:55	0:02:07	Syntax error	Altered syntax
32	CPLEX	4/28/2020 18:49	0:02:59	Syntax error	Altered syntax
31.4	CPLEX	4/28/2020 18:48	0:50:01	Program finished	Problem found with isolated loops, variables modified
31.3	CPLEX	4/28/2020 18:44	0:00:34	Syntax error	Altered syntax
31.2	CPLEX	4/28/2020 18:40	0:01:12	Syntax error	Altered syntax
31.1	CPLEX	4/28/2020 18:13	0:00:02	Syntax error	Altered syntax
31	CPLEX	4/28/2020 18:12	0:00:20	Syntax error	Altered syntax
30	feaspump	4/28/2020 5:46	0:00:05	Syntax error	The solver feaspump doesn't have necessary functions

29	FICO-Xpress	4/28/2020 5:45	8:00:31	Timed out	This number of variables is still a problem using feaspump
28	Gurobi	4/28/2020 5:44	8:00:23	Timed out	This number of variables is still a problem using Gurobi
27	MOSEK	4/28/2020 5:43	0:00:33	Syntax error	MOSEK doesn't have necessary functions
26.1	scip	4/28/2020 5:42	11:06:42	Timed out	This number of variables is still a problem using scip
26	scip	4/28/2020 5:40	0:00:29	Syntax error	Altered syntax
25.5	CPLEX	4/28/2020 5:22	0:00:24	Syntax error	Altered syntax
25.4	CPLEX	4/28/2020 4:54	0:00:08	Syntax error	Altered syntax
25.3	CPLEX	4/28/2020 4:53	0:00:02	Syntax error	Altered syntax
25.2	CPLEX	4/28/2020 4:45	0:00:07	Syntax error	Altered syntax
25.1	CPLEX	4/28/2020 4:38	0:00:02	Syntax error	Altered syntax
25	CPLEX	4/28/2020 4:31	0:00:10	Syntax error	Altered syntax
24.3	CPLEX	4/28/2020 0:03	8:00:15	Timed out	Number of variables must be reduced

24.2	CPLEX	4/28/2020 0:02	8:00:17	Timed out	Number of variables must be reduced
24.1	CPLEX	4/28/2020 0:01	8:00:30	Timed out	Number of variables must be reduced
24	CPLEX	4/28/2020 0:00	8:00:21	Timed out	Number of variables must be reduced
23.2	CPLEX	4/27/2020 22:10	1:36:09	Out of memory	Problem found with isolated loops, constraints modified
23.1	CPLEX	4/27/2020 22:00	1:45:48	Out of memory	Values in .dat file changed
23	CPLEX	4/27/2020 21:56	0:00:09	Syntax error	Altered syntax
22	CPLEX	4/27/2020 21:24	0:46:56	Program finished	Constraints modified
21	CPLEX	4/27/2020 21:03	0:16:44	Program finished	Constraints modified
20.3	CPLEX	4/27/2020 21:02	0:10:40	Program finished	Constraints modified
20.2	CPLEX	4/27/2020 20:39	0:08:49	Syntax error	Altered syntax
20.1	CPLEX	4/27/2020 17:08	2:22:13	Program finished	New variables broke old constraints
20	CPLEX	4/27/2020 16:50	0:00:02	Syntax error	Altered syntax

19	CPLEX	4/27/2020 4:57	0:15:33	Out of memory	Variable definitions changed
18.4	CPLEX	4/27/2020 3:47	0:16:08	Out of memory	Variable definitions changed
18.3	CPLEX	4/27/2020 2:51	0:51:54	Program finished	Lane constraint corrected
18.2	CPLEX	4/27/2020 1:52	0:52:26	Program finished	Lane constraint still not working
18.1	CPLEX	4/27/2020 0:28	0:00:22	Syntax error	Altered syntax
17.12	CPLEX	4/27/2020 0:03	0:25:11	Program finished	Flaw in lane constraint found
17.11	CPLEX	4/26/2020 23:53	0:00:59	Syntax error	Altered syntax
17.9	CPLEX	4/26/2020 23:51	0:00:20	Syntax error	Altered syntax
17.8	CPLEX	4/26/2020 23:45	0:00:18	Syntax error	Altered syntax
17.7	CPLEX	4/26/2020 23:16	0:00:40	Syntax error	Altered syntax
17.6	CPLEX	4/26/2020 23:10	0:00:38	Syntax error	Altered syntax
17.5	CPLEX	4/26/2020 21:51	0:02:02	Syntax error	Altered syntax

17.4	CPLEX	4/26/2020 21:31	0:00:18	Syntax error	Altered syntax
17.3	CPLEX	4/26/2020 21:29	0:00:15	Syntax error	Altered syntax
17.2	CPLEX	4/26/2020 20:11	0:01:39	Syntax error	Altered syntax
17.1	CPLEX	4/26/2020 20:00	0:01:00	Syntax error	Altered syntax
17	CPLEX	4/26/2020 19:57	0:00:01	Syntax error	Altered syntax

Appendix K - Final AMPL Code

```
param n;        #Number of nodes
param o;        #^
param p;        #Number of arcs between nodes
param q;        #Number of plows
param TURNPENALTY;

set I := {1..n};   #Set of nodes
set J := {1..o};   #^
set POSITIONS := {1..p};        #Position of an arc in a route
set PLOWS := {1..q};    #Set of plows



set ARCS := {(1, 2), (1, 3), (2, 1), (2, 4), (2, 5), (3, 1), (3, 4), (3, 7), (4, 2), (4, 3), (4, 5), (4, 8), (5, 2), (5,
4), (6, 9), (7, 3), (7, 12), (7, 24), (8, 4), (8, 9), (8, 13), (9, 6), (9, 8), (9, 14), (10, 34), (11, 12), (11, 21), (12,
11), (12, 7), (12, 23), (13, 8), (13, 14), (14, 9), (14, 13), (14, 18), (15, 35), (16, 20), (17, 20), (17, 70), (18,
14), (18, 28), (18, 29), (19, 31), (19, 33), (20, 16), (20, 17), (20, 36), (21, 11), (21, 22), (22, 21), (22, 23),
(22, 52), (23, 12), (23, 22), (23, 24), (23, 53), (24, 7), (24, 23), (24, 25), (24, 54), (25, 24), (25, 26), (25,
40), (26, 13), (26, 25), (26, 27), (26, 37), (27, 26), (27, 28), (27, 37), (28, 18), (28, 27), (28, 29), (29, 18),
(29, 28), (29, 39), (29, 42), (30, 31), (30, 38), (31, 19), (31, 30), (31, 32), (32, 31), (32, 33), (32, 38), (33,
19), (33, 32), (33, 34), (34, 10), (34, 33), (34, 35), (35, 15), (35, 34), (35, 36), (36, 20), (36, 35), (36, 43),
(37, 27), (37, 26), (37, 41), (38, 30), (38, 32), (38, 49), (39, 29), (39, 48), (39, 49), (40, 25), (40, 41), (40,
55), (41, 37), (41, 40), (41, 42), (41, 57), (42, 29), (42, 41), (43, 36), (43, 44), (43, 45), (44, 43), (44, 45),
(44, 62), (45, 43), (45, 44), (46, 47), (46, 58), (47, 46), (47, 48), (47, 59), (48, 39), (48, 47), (48, 60), (49,
38), (49, 39), (49, 61), (50, 51), (50, 73), (51, 50), (51, 52), (51, 74), (52, 22), (52, 51), (52, 75), (53, 23),
(53, 52), (53, 54), (53, 76), (54, 24), (54, 53), (54, 55), (55, 40), (55, 54), (55, 56), (55, 77), (56, 55), (56,
57), (56, 67), (57, 41), (57, 56), (57, 58), (57, 80), (58, 46), (58, 57), (58, 59), (58, 81), (59, 47), (59, 58),
(59, 60), (60, 48), (60, 59), (60, 61), (61, 49), (61, 60), (61, 62), (61, 64), (62, 44), (62, 61), (62, 63), (62,
270), (63, 62), (64, 61), (64, 66), (64, 69), (65, 66), (66, 64), (66, 65), (66, 68), (67, 56), (67, 78), (67, 79),
(68, 66), (68, 69), (68, 71), (69, 64), (69, 68), (69, 270), (69, 278), (70, 17), (70, 88), (70, 270), (71, 68),
(71, 83), (71, 278), (72, 73), (73, 50), (73, 72), (73, 74), (74, 51), (74, 73), (74, 75), (74, 272), (75, 52),
(75, 74), (75, 76), (76, 53), (76, 75), (76, 77), (76, 147), (77, 55), (77, 76), (77, 78), (78, 67), (78, 77), (78,
79), (79, 67), (79, 78), (79, 80), (80, 57), (80, 79), (80, 81), (80, 273), (81, 58), (81, 80), (81, 82), (81,
100), (82, 81), (82, 83), (82, 93), (83, 71), (83, 82), (83, 84), (83, 89), (84, 83), (84, 85), (84, 90), (85, 84),
(85, 86), (85, 91), (85, 278), (86, 85), (86, 87), (86, 91), (87, 86), (87, 271), (87, 97), (88, 70), (88, 271),
(88, 116), (88, 131), (89, 83), (89, 90), (89, 94), (90, 84), (90, 89), (90, 91), (90, 95), (91, 85), (91, 86),
(91, 90), (91, 96), (92, 272), (93, 82), (93, 102), (93, 94), (94, 89), (94, 93), (94, 95), (94, 103), (95, 90),
(95, 94), (95, 96), (96, 91), (96, 95), (96, 97), (96, 107), (97, 87), (97, 96), (97, 110), (98, 78), (98, 99),
(98, 279), (99, 98), (99, 132), (99, 273), (100, 81), (100, 101), (100, 134), (100, 273), (101, 100), (102,
93), (102, 103), (102, 274), (103, 94), (103, 275), (103, 102), (104, 274), (104, 105), (104, 118), (105,
104), (105, 106), (105, 119), (106, 105), (106, 275), (106, 120), (107, 96), (107, 275), (107, 108), (107,
121), (108, 107), (108, 109), (108, 122), (109, 108), (109, 110), (109, 123), (110, 97), (110, 109), (110,
111), (110, 124), (111, 112), (111, 110), (111, 125), (112, 111), (112, 113), (112, 126), (113, 112), (113,
114), (113, 127), (114, 113), (114, 115), (114, 128), (114, 271), (115, 114), (115, 116), (115, 129), (116,
```

88), (116, 115), (116, 130), (117, 118), (117, 135), (117, 274), (118, 104), (118, 117), (118, 137), (119, 105), (119, 118), (119, 139), (120, 106), (120, 119), (120, 141), (121, 107), (121, 120), (121, 158), (122, 108), (122, 121), (122, 160), (123, 109), (123, 122), (123, 161), (124, 110), (124, 123), (124, 162), (125, 111), (125, 124), (125, 163), (126, 112), (126, 125), (126, 164), (127, 113), (127, 126), (127, 165), (128, 114), (128, 127), (128, 129), (128, 166), (129, 115), (129, 128), (129, 130), (129, 167), (130, 116), (130, 129), (130, 131), (130, 142), (131, 88), (131, 130), (131, 143), (132, 99), (132, 133), (132, 152), (133, 273), (133, 132), (133, 134), (133, 153), (134, 100), (134, 133), (134, 135), (135, 117), (135, 134), (135, 136), (135, 154), (136, 135), (136, 137), (136, 155), (137, 136), (137, 118), (137, 138), (138, 137), (138, 139), (138, 156), (139, 119), (139, 138), (139, 140), (140, 139), (140, 141), (140, 157), (141, 120), (141, 140), (142, 130), (142, 143), (142, 168), (143, 131), (143, 142), (143, 169), (144, 145), (145, 144), (145, 146), (145, 187), (146, 272), (146, 145), (146, 147), (146, 187), (147, 76), (147, 146), (147, 148), (147, 276), (148, 147), (148, 279), (148, 189), (149, 279), (149, 152), (149, 150), (149, 151), (150, 151), (150, 149), (150, 192), (151, 149), (151, 193), (152, 149), (152, 132), (152, 153), (153, 133), (153, 152), (153, 154), (153, 194), (154, 153), (154, 135), (154, 171), (154, 155), (155, 136), (155, 154), (155, 156), (155, 172), (156, 138), (156, 155), (156, 157), (156, 173), (157, 156), (157, 158), (157, 140), (157, 174), (158, 121), (158, 159), (159, 160), (159, 175), (160, 122), (160, 161), (161, 123), (161, 162), (161, 176), (162, 124), (162, 163), (162, 177), (163, 125), (163, 164), (163, 178), (164, 126), (164, 165), (164, 179), (165, 127), (165, 166), (165, 180), (166, 128), (166, 167), (166, 181), (167, 168), (167, 129), (167, 182), (168, 142), (168, 183), (169, 143), (169, 170), (169, 207), (170, 169), (170, 208), (171, 154), (171, 172), (171, 197), (172, 155), (172, 171), (172, 173), (173, 156), (173, 172), (173, 174), (174, 157), (174, 173), (174, 175), (174, 198), (175, 159), (175, 174), (175, 176), (176, 161), (176, 175), (176, 177), (177, 162), (177, 176), (177, 178), (177, 202), (178, 163), (178, 177), (178, 179), (179, 164), (179, 178), (179, 180), (180, 165), (180, 179), (180, 181), (181, 166), (181, 180), (181, 182), (182, 167), (182, 181), (182, 183), (182, 184), (183, 168), (183, 182), (183, 185), (184, 182), (184, 185), (184, 206), (185, 183), (185, 184), (185, 186), (186, 185), (187, 145), (187, 146), (187, 209), (187, 276), (188, 189), (188, 276), (189, 148), (189, 188), (189, 211), (190, 191), (191, 190), (191, 192), (191, 211), (192, 150), (192, 191), (192, 212), (193, 151), (193, 194), (193, 213), (194, 193), (194, 153), (194, 195), (194, 196), (195, 194), (195, 214), (195, 215), (196, 194), (196, 197), (196, 216), (197, 171), (197, 196), (197, 198), (197, 217), (198, 174), (198, 197), (198, 199), (199, 198), (199, 200), (199, 201), (200, 199), (200, 201), (201, 199), (201, 200), (201, 218), (202, 177), (202, 203), (202, 204), (203, 202), (203, 205), (203, 219), (204, 202), (204, 205), (204, 206), (205, 203), (205, 204), (205, 220), (206, 184), (206, 204), (206, 252), (207, 169), (207, 237), (207, 208), (207, 236), (208, 170), (208, 207), (208, 255), (209, 187), (209, 210), (209, 238), (210, 276), (210, 209), (210, 211), (211, 189), (211, 230), (212, 191), (212, 192), (212, 213), (213, 193), (213, 212), (213, 214), (213, 222), (214, 195), (214, 213), (214, 215), (214, 231), (215, 214), (215, 195), (215, 216), (216, 196), (216, 215), (216, 217), (217, 197), (217, 216), (217, 218), (218, 201), (218, 217), (218, 219), (218, 224), (219, 203), (219, 218), (219, 220), (219, 226), (220, 205), (220, 219), (220, 221), (220, 227), (221, 220), (222, 213), (222, 223), (222, 230), (223, 222), (224, 218), (224, 225), (225, 224), (225, 228), (225, 281), (226, 219), (226, 227), (226, 228), (227, 220), (227, 226), (227, 229), (228, 225), (228, 226), (228, 229), (228, 232), (229, 227), (229, 228), (229, 233), (230, 211), (230, 222), (230, 231), (231, 214), (231, 230), (231, 239), (232, 228), (232, 233), (232, 235), (232, 281), (233, 229), (233, 232), (233, 234), (234, 233), (235, 232), (235, 251), (235, 281), (236, 207), (236, 237), (236, 280), (237, 207), (237, 236), (238, 209), (239, 231), (239, 240), (239, 243), (240, 239), (240, 241), (241, 240), (241, 250), (242, 243), (242, 246), (242, 247), (243, 244), (243, 239), (243, 242), (243, 248), (244, 243), (245, 246), (246, 245), (246, 242), (246, 247), (247, 242), (247, 246), (247, 248), (248, 243), (248, 247), (248, 249), (248, 263), (249, 240), (249, 248), (249, 250), (250, 241), (250, 249), (250, 251), (250, 256), (251, 235), (251, 250), (251, 252), (251, 259), (252, 206), (252, 251), (252, 255), (253, 280), (253, 254), (253, 255), (254, 280), (254, 253), (255, 208), (255, 252), (255, 253), (256, 250), (256, 257), (256, 264), (257, 256), (258, 259), (259, 251), (259, 258), (259, 260), (260, 259), (260, 277), (261, 262), (262, 261), (262, 263), (263, 248), (263,

```
262), (263, 267), (264, 256), (264, 263), (264, 265), (265, 264), (265, 266), (265, 268), (266, 259), (266,
265), (266, 277), (267, 263), (268, 265), (269, 277), (270, 62), (270, 69), (270, 70), (270, 271), (271,
270), (271, 87), (271, 88), (271, 114), (272, 92), (272, 74), (272, 146), (273, 80), (273, 99), (273, 100),
(273, 133), (274, 102), (274, 104), (274, 117), (275, 103), (275, 106), (275, 107), (276, 147), (276, 187),
(276, 188), (276, 210), (277, 260), (277, 266), (277, 269), (278, 69), (278, 71), (278, 85), (279, 98), (279,
148), (279, 149), (280, 236), (280, 253), (280, 254), (281, 225), (281, 232), (281, 235)};   #Set of arcs
between nodes

set GRADERARCS := {(87, 97), (97, 110), (93, 94), (102, 103), (274, 104), (104, 105), (104, 118), (105,
106), (105, 119), (106, 275), (106, 120), (275, 107), (107, 108), (108, 122), (109, 123), (109, 110), (110,
124), (110, 111), (111, 112), (111, 125), (112, 113), (112, 126), (113, 114), (113, 127), (114, 128), (114,
115), (115, 129), (115, 116), (116, 130), (130, 131), (130, 142), (130, 129), (129, 128), (129, 167), (128,
166), (128, 127), (127, 165), (127, 126), (126, 164), (126, 125), (125, 163), (125, 124), (124, 162), (124,
123), (123, 161), (123, 122), (122, 160), (122, 121), (121, 158), (121, 120), (120, 141), (120, 119), (119,
139), (119, 118), (118, 137), (118, 117), (136, 137), (136, 155), (137, 138), (138, 156), (138, 139), (139,
140), (140, 157), (140, 141), (142, 168), (168, 183), (154, 155), (155, 172), (155, 156), (156, 173), (156,
157), (157, 174), (157, 158), (158, 159), (159, 175), (159, 160), (160, 161), (161, 176), (161, 162), (162,
177), (162, 163), (163, 178), (163, 164), (164, 179), (164, 165), (165, 180), (165, 166), (166, 181), (166,
167), (167, 182), (167, 168), (171, 172), (172, 173), (173, 174), (174, 175), (175, 176), (176, 177), (177,
202), (177, 178), (178, 179), (179, 180), (180, 181), (181, 182), (182, 183), (183, 185), (185, 184), (185,
186), (202, 204), (202, 203), (204, 206), (204, 205), (203, 205), (203, 219), (205, 220), (219, 226), (220,
227), (226, 227), (226, 228), (227, 229), (228, 229), (229, 233), (224, 225), (225, 281), (225, 228), (228,
232), (281, 232), (232, 233), (233, 234), (232, 235), (235, 281), (235, 251)};        #Set of grader arcs

param COST {I, J} >= 0;                          #Cost matrix
param PASSES {I, J} >= 0 integer;        #Number of passes required between nodes i and j

var T {ARCS, POSITIONS, PLOWS} binary;    #Decision variables

minimize z:  (sum {(i, j) in ARCS, k in POSITIONS, l in PLOWS} T[i, j, k, l] * COST[i, j])
   +
   TURNPENALTY * sum {k in POSITIONS, l in PLOWS} (T[1, 2, k, l] * T[2, 4, k, l] + T[4, 2, k, l] *
T[2, 1, k, l] + T[1, 3, k, l] * T[3, 4, k, l] + T[4, 3, k, l] * T[3, 1, k, l] + T[2, 1, k, l] * T[1, 3, k, l] + T[3, 1,
k, l] * T[1, 2, k, l] + T[2, 4, k, l] * T[4, 3, k, l] + T[3, 4, k, l] * T[4, 2, k, l] + T[2, 4, k, l] * T[4, 5, k, l] +
T[5, 4, k, l] * T[4, 2, k, l] + T[3, 4, k, l] * T[4, 8, k, l] + T[8, 4, k, l] * T[4, 3, k, l] + T[3, 7, k, l] * T[7, 12,
k, l] + T[12, 7, k, l] * T[7, 3, k, l] + T[4, 8, k, l] * T[8, 13, k, l] + T[13, 8, k, l] * T[8, 4, k, l] + T[5, 4, k, l]
* T[4, 8, k, l] + T[8, 4, k, l] * T[4, 5, k, l] + T[6, 9, k, l] * T[9, 8, k, l] + T[8, 9, k, l] * T[9, 6, k, l] + T[7,
12, k, l] * T[12, 11, k, l] + T[11, 12, k, l] * T[12, 7, k, l] + T[7, 24, k, l] * T[24, 23, k, l] + T[23, 24, k, l] *
T[24, 7, k, l] + T[7, 24, k, l] * T[24, 25, k, l] + T[25, 24, k, l] * T[24, 7, k, l] + T[8, 13, k, l] * T[13, 14, k,
l] + T[14, 13, k, l] * T[13, 8, k, l] + T[8, 9, k, l] * T[9, 14, k, l] + T[14, 9, k, l] * T[9, 8, k, l] + T[9, 8, k, l]
* T[8, 13, k, l] + T[13, 8, k, l] * T[8, 9, k, l] + T[9, 14, k, l] * T[14, 13, k, l] + T[13, 14, k, l] * T[14, 9, k,
l] + T[10, 34, k, l] * T[34, 33, k, l] + T[33, 34, k, l] * T[34, 10, k, l] + T[10, 34, k, l] * T[34, 35, k, l] +
T[35, 34, k, l] * T[34, 10, k, l] + T[11, 21, k, l] * T[21, 22, k, l] + T[22, 21, k, l] * T[21, 11, k, l] + T[11,
12, k, l] * T[12, 23, k, l] + T[23, 12, k, l] * T[12, 11, k, l] + T[12, 23, k, l] * T[23, 22, k, l] + T[22, 23, k,
l] * T[23, 12, k, l] + T[12, 23, k, l] * T[23, 24, k, l] + T[24, 23, k, l] * T[23, 12, k, l] + T[12, 7, k, l] * T[7,
24, k, l] + T[24, 7, k, l] * T[7, 12, k, l] + T[13, 14, k, l] * T[14, 18, k, l] + T[18, 14, k, l] * T[14, 13, k, l] +
T[14, 18, k, l] * T[18, 29, k, l] + T[29, 18, k, l] * T[18, 14, k, l] + T[15, 35, k, l] * T[35, 34, k, l] + T[34,
35, k, l] * T[35, 15, k, l] + T[15, 35, k, l] * T[35, 36, k, l] + T[36, 35, k, l] * T[35, 15, k, l] + T[16, 20, k,
```

l] * T[20, 17, k, l] + T[17, 20, k, l] * T[20, 16, k, l] + T[16, 20, k, l] * T[20, 36, k, l] + T[36, 20, k, l] * T[20, 16, k, l] + T[17, 70, k, l] * T[70, 270, k, l] + T[270, 70, k, l] * T[70, 17, k, l] + T[18, 29, k, l] * T[29, 28, k, l] + T[28, 29, k, l] * T[29, 18, k, l] + T[18, 28, k, l] * T[28, 27, k, l] + T[27, 28, k, l] * T[28, 18, k, l] + T[19, 31, k, l] * T[31, 30, k, l] + T[30, 31, k, l] * T[31, 19, k, l] + T[19, 31, k, l] * T[31, 32, k, l] + T[32, 31, k, l] * T[31, 19, k, l] + T[19, 33, k, l] * T[33, 32, k, l] + T[32, 33, k, l] * T[33, 19, k, l] + T[19, 33, k, l] * T[33, 34, k, l] + T[34, 33, k, l] * T[33, 19, k, l] + T[20, 17, k, l] * T[17, 70, k, l] + T[70, 17, k, l] * T[17, 20, k, l] + T[20, 36, k, l] * T[36, 35, k, l] + T[35, 36, k, l] * T[36, 20, k, l] + T[21, 22, k, l] * T[22, 52, k, l] + T[52, 22, k, l] * T[22, 21, k, l] + T[22, 23, k, l] * T[23, 53, k, l] + T[53, 23, k, l] * T[23, 22, k, l] + T[22, 52, k, l] * T[52, 51, k, l] + T[51, 52, k, l] * T[52, 22, k, l] + T[23, 24, k, l] * T[24, 54, k, l] + T[54, 24, k, l] * T[24, 23, k, l] + T[24, 23, k, l] * T[23, 53, k, l] + T[53, 23, k, l] * T[23, 24, k, l] + T[24, 25, k, l] * T[25, 40, k, l] + T[40, 25, k, l] * T[25, 24, k, l] + T[24, 54, k, l] * T[54, 53, k, l] + T[53, 54, k, l] * T[54, 24, k, l] + T[25, 24, k, l] * T[24, 54, k, l] + T[54, 24, k, l] * T[24, 25, k, l] + T[25, 40, k, l] * T[40, 41, k, l] + T[41, 40, k, l] * T[40, 25, k, l] + T[25, 26, k, l] * T[26, 37, k, l] + T[37, 26, k, l] * T[26, 25, k, l] + T[26, 27, k, l] * T[27, 37, k, l] + T[37, 27, k, l] * T[27, 26, k, l] + T[26, 37, k, l] * T[37, 27, k, l] + T[27, 37, k, l] * T[37, 26, k, l] + T[29, 42, k, l] * T[42, 41, k, l] + T[41, 42, k, l] * T[42, 29, k, l] + T[29, 39, k, l] * T[39, 48, k, l] + T[48, 39, k, l] * T[39, 29, k, l] + T[30, 38, k, l] * T[38, 32, k, l] + T[32, 38, k, l] * T[38, 30, k, l] + T[31, 32, k, l] * T[32, 38, k, l] + T[38, 32, k, l] * T[32, 31, k, l] + T[31, 30, k, l] * T[30, 38, k, l] + T[38, 30, k, l] * T[30, 31, k, l] + T[31, 19, k, l] * T[19, 33, k, l] + T[33, 19, k, l] * T[19, 31, k, l] + T[35, 36, k, l] * T[36, 43, k, l] + T[43, 36, k, l] * T[36, 35, k, l] + T[36, 43, k, l] * T[43, 45, k, l] + T[45, 43, k, l] * T[43, 36, k, l] + T[37, 41, k, l] * T[41, 40, k, l] + T[40, 41, k, l] * T[41, 37, k, l] + T[37, 41, k, l] * T[41, 42, k, l] + T[42, 41, k, l] * T[41, 37, k, l] + T[38, 49, k, l] * T[49, 39, k, l] + T[39, 49, k, l] * T[49, 38, k, l] + T[39, 48, k, l] * T[48, 47, k, l] + T[47, 48, k, l] * T[48, 39, k, l] + T[39, 49, k, l] * T[49, 61, k, l] + T[61, 49, k, l] * T[49, 39, k, l] + T[40, 55, k, l] * T[55, 54, k, l] + T[54, 55, k, l] * T[55, 40, k, l] + T[40, 55, k, l] * T[55, 56, k, l] + T[56, 55, k, l] * T[55, 40, k, l] + T[40, 41, k, l] * T[41, 57, k, l] + T[57, 41, k, l] * T[41, 40, k, l] + T[41, 57, k, l] * T[57, 56, k, l] + T[56, 57, k, l] * T[57, 41, k, l] + T[41, 57, k, l] * T[57, 58, k, l] + T[58, 57, k, l] * T[57, 41, k, l] + T[42, 41, k, l] * T[41, 57, k, l] + T[57, 41, k, l] * T[41, 42, k, l] + T[43, 44, k, l] * T[44, 45, k, l] + T[45, 44, k, l] * T[44, 43, k, l] + T[43, 45, k, l] * T[45, 44, k, l] + T[44, 45, k, l] * T[45, 43, k, l] + T[45, 44, k, l] * T[44, 62, k, l] + T[62, 44, k, l] * T[44, 45, k, l] + T[44, 43, k, l] * T[43, 45, k, l] + T[45, 43, k, l] * T[43, 44, k, l] + T[44, 62, k, l] * T[62, 63, k, l] + T[63, 62, k, l] * T[62, 44, k, l] + T[44, 62, k, l] * T[62, 61, k, l] + T[61, 62, k, l] * T[62, 44, k, l] + T[46, 47, k, l] * T[47, 59, k, l] + T[59, 47, k, l] * T[47, 46, k, l] + T[46, 58, k, l] * T[58, 57, k, l] + T[57, 58, k, l] * T[58, 46, k, l] + T[46, 58, k, l] * T[58, 59, k, l] + T[59, 58, k, l] * T[58, 46, k, l] + T[47, 46, k, l] * T[46, 58, k, l] + T[58, 46, k, l] * T[46, 47, k, l] + T[47, 48, k, l] * T[48, 60, k, l] + T[60, 48, k, l] * T[48, 47, k, l] + T[47, 59, k, l] * T[59, 58, k, l] + T[58, 59, k, l] * T[59, 47, k, l] + T[47, 59, k, l] * T[59, 60, k, l] + T[60, 59, k, l] * T[59, 47, k, l] + T[48, 47, k, l] * T[47, 59, k, l] + T[59, 47, k, l] * T[47, 48, k, l] + T[48, 39, k, l] * T[39, 49, k, l] + T[49, 39, k, l] * T[39, 48, k, l] + T[48, 60, k, l] * T[60, 59, k, l] + T[59, 60, k, l] * T[60, 48, k, l] + T[48, 60, k, l] * T[60, 61, k, l] + T[61, 60, k, l] * T[60, 48, k, l] + T[49, 61, k, l] * T[61, 60, k, l] + T[60, 61, k, l] * T[61, 49, k, l] + T[49, 61, k, l] * T[61, 62, k, l] + T[62, 61, k, l] * T[61, 49, k, l] + T[50, 51, k, l] * T[51, 74, k, l] + T[74, 51, k, l] * T[51, 50, k, l] + T[50, 73, k, l] * T[73, 72, k, l] + T[72, 73, k, l] * T[73, 50, k, l] + T[50, 73, k, l] * T[73, 74, k, l] + T[74, 73, k, l] * T[73, 50, k, l] + T[51, 52, k, l] * T[52, 75, k, l] + T[75, 52, k, l] * T[52, 51, k, l] + T[51, 74, k, l] * T[74, 73, k, l] + T[73, 74, k, l] * T[74, 51, k, l] + T[51, 74, k, l] * T[74, 75, k, l] + T[75, 74, k, l] * T[74, 51, k, l] + T[51, 50, k, l] * T[50, 73, k, l] + T[73, 50, k, l] * T[50, 51, k, l] + T[52, 75, k, l] * T[75, 76, k, l] + T[76, 75, k, l] * T[75, 52, k, l] + T[52, 75, k, l] * T[75, 74, k, l] + T[74, 75, k, l] * T[75, 52, k, l] + T[52, 51, k, l] * T[51, 74, k, l] + T[74, 51, k, l] * T[51, 52, k, l] + T[54, 53, k, l] * T[53, 76, k, l] + T[76, 53, k, l] * T[53, 54, k, l] + T[54, 55, k, l] * T[55, 77, k, l] + T[77, 55, k, l] * T[55, 54, k, l] + T[56, 55, k, l] * T[55, 77, k, l] + T[77, 55, k, l] * T[55, 56, k, l] + T[56, 67, k, l] * T[67, 79, k, l] + T[79, 67, k, l] * T[67, 56, k, l] + T[56, 57, k, l] * T[57, 80, k, l] + T[80, 57, k, l] * T[57, 56, k, l] + T[57,

56, k, l] * T[56, 67, k, l] + T[67, 56, k, l] * T[56, 57, k, l] + T[57, 58, k, l] * T[58, 81, k, l] + T[81, 58, k, l] * T[58, 57, k, l] + T[57, 80, k, l] * T[80, 79, k, l] + T[79, 80, k, l] * T[80, 57, k, l] + T[57, 80, k, l] * T[80, 81, k, l] + T[81, 80, k, l] * T[80, 57, k, l] + T[58, 57, k, l] * T[57, 80, k, l] + T[80, 57, k, l] * T[57, 58, k, l] + T[58, 81, k, l] * T[81, 80, k, l] + T[80, 81, k, l] * T[81, 58, k, l] + T[58, 81, k, l] * T[81, 82, k, l] + T[82, 81, k, l] * T[81, 58, k, l] + T[59, 58, k, l] * T[58, 81, k, l] + T[81, 58, k, l] * T[58, 59, k, l] + T[60, 61, k, l] * T[61, 64, k, l] + T[64, 61, k, l] * T[61, 60, k, l] + T[61, 62, k, l] * T[62, 270, k, l] + T[270, 62, k, l] * T[62, 61, k, l] + T[61, 64, k, l] * T[64, 69, k, l] + T[69, 64, k, l] * T[64, 61, k, l] + T[62, 61, k, l] * T[61, 64, k, l] + T[64, 61, k, l] * T[61, 62, k, l] + T[62, 270, k, l] * T[270, 69, k, l] + T[69, 270, k, l] * T[270, 62, k, l] + T[62, 270, k, l] * T[270, 70, k, l] + T[70, 270, k, l] * T[270, 62, k, l] + T[63, 62, k, l] * T[62, 270, k, l] + T[270, 62, k, l] * T[62, 63, k, l] + T[64, 66, k, l] * T[66, 65, k, l] + T[65, 66, k, l] * T[66, 64, k, l] + T[64, 69, k, l] * T[69, 68, k, l] + T[68, 69, k, l] * T[69, 64, k, l] + T[64, 69, k, l] * T[69, 270, k, l] + T[270, 69, k, l] * T[69, 64, k, l] + T[65, 66, k, l] * T[66, 68, k, l] + T[68, 66, k, l] * T[66, 65, k, l] + T[66, 64, k, l] * T[64, 69, k, l] + T[69, 64, k, l] * T[64, 66, k, l] + T[66, 68, k, l] * T[68, 69, k, l] + T[69, 68, k, l] * T[68, 66, k, l] + T[67, 78, k, l] * T[78, 77, k, l] + T[77, 78, k, l] * T[78, 67, k, l] + T[67, 78, k, l] * T[78, 79, k, l] + T[79, 78, k, l] * T[78, 67, k, l] + T[67, 79, k, l] * T[79, 78, k, l] + T[78, 79, k, l] * T[79, 67, k, l] + T[67, 79, k, l] * T[79, 80, k, l] + T[80, 79, k, l] * T[79, 67, k, l] + T[68, 71, k, l] * T[71, 278, k, l] + T[278, 71, k, l] * T[71, 68, k, l] + T[68, 69, k, l] * T[69, 278, k, l] + T[278, 69, k, l] * T[69, 68, k, l] + T[69, 68, k, l] * T[68, 71, k, l] + T[71, 68, k, l] * T[68, 69, k, l] + T[69, 278, k, l] * T[278, 71, k, l] + T[71, 278, k, l] * T[278, 69, k, l] + T[70, 270, k, l] * T[270, 271, k, l] + T[271, 270, k, l] * T[270, 70, k, l] + T[70, 88, k, l] * T[88, 271, k, l] + T[271, 88, k, l] * T[88, 70, k, l] + T[70, 88, k, l] * T[88, 131, k, l] + T[131, 88, k, l] * T[88, 70, k, l] + T[71, 278, k, l] * T[278, 85, k, l] + T[85, 278, k, l] * T[278, 71, k, l] + T[71, 83, k, l] * T[83, 82, k, l] + T[82, 83, k, l] * T[83, 71, k, l] + T[71, 83, k, l] * T[83, 84, k, l] + T[84, 83, k, l] * T[83, 71, k, l] + T[73, 74, k, l] * T[74, 272, k, l] + T[272, 74, k, l] * T[74, 73, k, l] + T[74, 272, k, l] * T[272, 92, k, l] + T[92, 272, k, l] * T[272, 74, k, l] + T[75, 74, k, l] * T[74, 272, k, l] + T[272, 74, k, l] * T[74, 75, k, l] + T[75, 76, k, l] * T[76, 147, k, l] + T[147, 76, k, l] * T[76, 75, k, l] + T[76, 147, k, l] * T[147, 146, k, l] + T[146, 147, k, l] * T[147, 76, k, l] + T[76, 147, k, l] * T[147, 148, k, l] + T[148, 147, k, l] * T[147, 76, k, l] + T[77, 76, k, l] * T[76, 147, k, l] + T[147, 76, k, l] * T[76, 77, k, l] + T[78, 67, k, l] * T[67, 79, k, l] + T[79, 67, k, l] * T[67, 78, k, l] + T[79, 80, k, l] * T[80, 273, k, l] + T[273, 80, k, l] * T[80, 79, k, l] + T[80, 273, k, l] * T[273, 99, k, l] + T[99, 273, k, l] * T[273, 80, k, l] + T[80, 273, k, l] * T[273, 100, k, l] + T[100, 273, k, l] * T[273, 80, k, l] + T[80, 81, k, l] * T[81, 100, k, l] + T[100, 81, k, l] * T[81, 80, k, l] + T[81, 100, k, l] * T[100, 273, k, l] + T[273, 100, k, l] * T[100, 81, k, l] + T[81, 100, k, l] * T[100, 101, k, l] + T[101, 100, k, l] * T[100, 81, k, l] + T[81, 82, k, l] * T[82, 93, k, l] + T[93, 82, k, l] * T[82, 81, k, l] + T[82, 81, k, l] * T[81, 100, k, l] + T[100, 81, k, l] * T[81, 82, k, l] + T[82, 83, k, l] * T[83, 89, k, l] + T[89, 83, k, l] * T[83, 82, k, l] + T[82, 93, k, l] * T[93, 94, k, l] + T[94, 93, k, l] * T[93, 82, k, l] + T[83, 82, k, l] * T[82, 93, k, l] + T[93, 82, k, l] * T[82, 83, k, l] + T[83, 84, k, l] * T[84, 90, k, l] + T[90, 84, k, l] * T[84, 83, k, l] + T[83, 89, k, l] * T[89, 90, k, l] + T[90, 89, k, l] * T[89, 83, k, l] + T[84, 83, k, l] * T[83, 89, k, l] + T[89, 83, k, l] * T[83, 84, k, l] + T[84, 85, k, l] * T[85, 91, k, l] + T[91, 85, k, l] * T[85, 84, k, l] + T[84, 90, k, l] * T[90, 89, k, l] + T[89, 90, k, l] * T[90, 84, k, l] + T[84, 90, k, l] * T[90, 91, k, l] + T[91, 90, k, l] * T[90, 84, k, l] + T[85, 86, k, l] * T[86, 91, k, l] + T[91, 86, k, l] * T[86, 85, k, l] + T[85, 84, k, l] * T[84, 90, k, l] + T[90, 84, k, l] * T[84, 85, k, l] + T[85, 91, k, l] * T[91, 86, k, l] + T[86, 91, k, l] * T[91, 85, k, l] + T[85, 91, k, l] * T[91, 90, k, l] + T[90, 91, k, l] * T[91, 85, k, l] + T[86, 87, k, l] * T[87, 97, k, l] + T[97, 87, k, l] * T[87, 86, k, l] + T[86, 85, k, l] * T[85, 278, k, l] + T[278, 85, k, l] * T[85, 86, k, l] + T[86, 85, k, l] * T[85, 91, k, l] + T[91, 85, k, l] * T[85, 86, k, l] + T[87, 86, k, l] * T[86, 91, k, l] + T[91, 86, k, l] * T[86, 87, k, l] + T[87, 271, k, l] * T[271, 270, k, l] + T[270, 271, k, l] * T[271, 87, k, l] + T[87, 271, k, l] * T[271, 114, k, l] + T[114, 271, k, l] * T[271, 87, k, l] + T[88, 271, k, l] * T[271, 270, k, l] + T[270, 271, k, l] * T[271, 88, k, l] + T[88, 70, k, l] * T[70, 270, k, l] + T[270, 70, k, l] * T[70, 88, k, l] + T[88, 131, k, l] * T[131, 130, k, l] + T[130, 131, k, l] * T[131, 88, k, l] + T[88, 116, k, l] * T[116, 115, k, l] + T[115, 116, k, l] * T[116, 88, k, l] +

T[89, 94, k, l] * T[94, 93, k, l] + T[93, 94, k, l] * T[94, 89, k, l] + T[89, 94, k, l] * T[94, 95, k, l] + T[95, 94, k, l] * T[94, 89, k, l] + T[89, 90, k, l] * T[90, 95, k, l] + T[95, 90, k, l] * T[90, 89, k, l] + T[90, 89, k, l] * T[89, 94, k, l] + T[94, 89, k, l] * T[89, 90, k, l] + T[90, 95, k, l] * T[95, 94, k, l] + T[94, 95, k, l] * T[95, 90, k, l] + T[90, 95, k, l] * T[95, 96, k, l] + T[96, 95, k, l] * T[95, 90, k, l] + T[90, 91, k, l] * T[91, 96, k, l] + T[96, 91, k, l] * T[91, 90, k, l] + T[91, 90, k, l] * T[90, 95, k, l] + T[95, 90, k, l] * T[90, 91, k, l] + T[91, 96, k, l] * T[96, 97, k, l] + T[97, 96, k, l] * T[96, 91, k, l] + T[91, 96, k, l] * T[96, 95, k, l] + T[95, 96, k, l] * T[96, 91, k, l] + T[92, 272, k, l] * T[272, 146, k, l] + T[146, 272, k, l] * T[272, 92, k, l] + T[93, 94, k, l] * T[94, 103, k, l] + T[103, 94, k, l] * T[94, 93, k, l] + T[93, 102, k, l] * T[102, 103, k, l] + T[103, 102, k, l] * T[102, 93, k, l] + T[94, 93, k, l] * T[93, 102, k, l] + T[102, 93, k, l] * T[93, 94, k, l] + T[94, 103, k, l] * T[103, 102, k, l] + T[102, 103, k, l] * T[103, 94, k, l] + T[95, 96, k, l] * T[96, 107, k, l] + T[107, 96, k, l] * T[96, 95, k, l] + T[96, 97, k, l] * T[97, 110, k, l] + T[110, 97, k, l] * T[97, 96, k, l] + T[97, 96, k, l] * T[96, 107, k, l] + T[107, 96, k, l] * T[96, 97, k, l] + T[97, 110, k, l] * T[110, 109, k, l] + T[109, 110, k, l] * T[110, 97, k, l] + T[97, 110, k, l] * T[110, 111, k, l] + T[111, 110, k, l] * T[110, 97, k, l] + T[98, 99, k, l] * T[99, 132, k, l] + T[132, 99, k, l] * T[99, 98, k, l] + T[99, 132, k, l] * T[132, 152, k, l] + T[152, 132, k, l] * T[132, 99, k, l] + T[99, 132, k, l] * T[132, 133, k, l] + T[133, 132, k, l] * T[132, 99, k, l] + T[99, 273, k, l] * T[273, 133, k, l] + T[133, 273, k, l] * T[273, 99, k, l] + T[100, 273, k, l] * T[273, 133, k, l] + T[133, 273, k, l] * T[273, 100, k, l] + T[100, 134, k, l] * T[134, 133, k, l] + T[133, 134, k, l] * T[134, 100, k, l] + T[100, 134, k, l] * T[134, 135, k, l] + T[135, 134, k, l] * T[134, 100, k, l] + T[101, 100, k, l] * T[100, 134, k, l] + T[134, 100, k, l] * T[100, 101, k, l] + T[102, 103, k, l] * T[103, 275, k, l] + T[275, 103, k, l] * T[103, 102, k, l] + T[102, 274, k, l] * T[274, 104, k, l] + T[104, 274, k, l] * T[274, 102, k, l] + T[103, 102, k, l] * T[102, 274, k, l] + T[274, 102, k, l] * T[102, 103, k, l] + T[103, 275, k, l] * T[275, 106, k, l] + T[106, 275, k, l] * T[275, 103, k, l] + T[103, 275, k, l] * T[275, 107, k, l] + T[107, 275, k, l] * T[275, 103, k, l] + T[104, 274, k, l] * T[274, 117, k, l] + T[117, 274, k, l] * T[274, 104, k, l] + T[104, 118, k, l] * T[118, 117, k, l] + T[117, 118, k, l] * T[118, 104, k, l] + T[104, 105, k, l] * T[105, 119, k, l] + T[119, 105, k, l] * T[105, 104, k, l] + T[105, 104, k, l] * T[104, 118, k, l] + T[118, 104, k, l] * T[104, 105, k, l] + T[105, 106, k, l] * T[106, 120, k, l] + T[120, 106, k, l] * T[106, 105, k, l] + T[106, 105, k, l] * T[105, 119, k, l] + T[119, 105, k, l] * T[105, 106, k, l] + T[107, 108, k, l] * T[108, 122, k, l] + T[122, 108, k, l] * T[108, 107, k, l] + T[108, 107, k, l] * T[107, 121, k, l] + T[121, 107, k, l] * T[107, 108, k, l] + T[108, 109, k, l] * T[109, 123, k, l] + T[123, 109, k, l] * T[109, 108, k, l] + T[109, 108, k, l] * T[108, 122, k, l] + T[122, 108, k, l] * T[108, 109, k, l] + T[110, 109, k, l] * T[109, 123, k, l] + T[123, 109, k, l] * T[109, 110, k, l] + T[110, 111, k, l] * T[111, 125, k, l] + T[125, 111, k, l] * T[111, 110, k, l] + T[111, 110, k, l] * T[110, 124, k, l] + T[124, 110, k, l] * T[110, 111, k, l] + T[111, 112, k, l] * T[112, 126, k, l] + T[126, 112, k, l] * T[112, 111, k, l] + T[112, 111, k, l] * T[111, 125, k, l] + T[125, 111, k, l] * T[111, 112, k, l] + T[112, 113, k, l] * T[113, 127, k, l] + T[127, 113, k, l] * T[113, 112, k, l] + T[113, 112, k, l] * T[112, 126, k, l] + T[126, 112, k, l] * T[112, 113, k, l] + T[113, 114, k, l] * T[114, 128, k, l] + T[128, 114, k, l] * T[114, 113, k, l] + T[114, 113, k, l] * T[113, 127, k, l] + T[127, 113, k, l] * T[113, 114, k, l] + T[114, 128, k, l] * T[128, 129, k, l] + T[129, 128, k, l] * T[128, 114, k, l] + T[114, 115, k, l] * T[115, 129, k, l] + T[129, 115, k, l] * T[115, 114, k, l] + T[115, 114, k, l] * T[114, 128, k, l] + T[128, 114, k, l] * T[114, 115, k, l] + T[115, 129, k, l] * T[129, 128, k, l] + T[128, 129, k, l] * T[129, 115, k, l] + T[115, 129, k, l] * T[129, 130, k, l] + T[130, 129, k, l] * T[129, 115, k, l] + T[115, 116, k, l] * T[116, 130, k, l] + T[130, 116, k, l] * T[116, 115, k, l] + T[116, 115, k, l] * T[115, 129, k, l] + T[129, 115, k, l] * T[115, 116, k, l] + T[116, 130, k, l] * T[130, 129, k, l] + T[129, 130, k, l] * T[130, 116, k, l] + T[116, 130, k, l] * T[130, 131, k, l] + T[131, 130, k, l] * T[130, 116, k, l] + T[117, 135, k, l] * T[135, 136, k, l] + T[136, 135, k, l] * T[135, 117, k, l] + T[118, 137, k, l] * T[137, 136, k, l] + T[136, 137, k, l] * T[137, 118, k, l] + T[118, 117, k, l] * T[117, 135, k, l] + T[135, 117, k, l] * T[117, 118, k, l] + T[118, 137, k, l] * T[137, 138, k, l] + T[138, 137, k, l] * T[137, 118, k, l] + T[119, 139, k, l] * T[139, 138, k, l] + T[138, 139, k, l] * T[139, 119, k, l] + T[119, 139, k, l] * T[139, 140, k, l] + T[140, 139, k, l] * T[139, 119, k, l] + T[120, 141, k, l] * T[141, 140, k, l] + T[140, 141, k, l] * T[141, 120, k, l] + T[128, 129, k, l] *

T[129, 167, k, l] + T[167, 129, k, l] * T[129, 128, k, l] + T[129, 128, k, l] * T[128, 166, k, l] + T[166, 128, k, l] * T[128, 129, k, l] + T[129, 130, k, l] * T[130, 142, k, l] + T[142, 130, k, l] * T[130, 129, k, l] + T[130, 129, k, l] * T[129, 167, k, l] + T[167, 129, k, l] * T[129, 130, k, l] + T[130, 142, k, l] * T[142, 143, k, l] + T[143, 142, k, l] * T[142, 130, k, l] + T[130, 131, k, l] * T[131, 143, k, l] + T[143, 131, k, l] * T[131, 130, k, l] + T[131, 130, k, l] * T[130, 142, k, l] + T[142, 130, k, l] * T[130, 131, k, l] + T[131, 143, k, l] * T[143, 169, k, l] + T[169, 143, k, l] * T[143, 131, k, l] + T[131, 143, k, l] * T[143, 142, k, l] + T[142, 143, k, l] * T[143, 131, k, l] + T[132, 99, k, l] * T[99, 273, k, l] + T[273, 99, k, l] * T[99, 132, k, l] + T[132, 133, k, l] * T[133, 273, k, l] + T[273, 133, k, l] * T[133, 132, k, l] + T[132, 133, k, l] * T[133, 153, k, l] + T[153, 133, k, l] * T[133, 132, k, l] + T[132, 152, k, l] * T[152, 153, k, l] + T[153, 152, k, l] * T[152, 132, k, l] + T[133, 153, k, l] * T[153, 154, k, l] + T[154, 153, k, l] * T[153, 133, k, l] + T[134, 133, k, l] * T[133, 153, k, l] + T[153, 133, k, l] * T[133, 134, k, l] + T[134, 135, k, l] * T[135, 154, k, l] + T[154, 135, k, l] * T[135, 134, k, l] + T[135, 136, k, l] * T[136, 155, k, l] + T[155, 136, k, l] * T[136, 135, k, l] + T[135, 154, k, l] * T[154, 153, k, l] + T[153, 154, k, l] * T[154, 135, k, l] + T[135, 154, k, l] * T[154, 155, k, l] + T[155, 154, k, l] * T[154, 135, k, l] + T[136, 155, k, l] * T[155, 154, k, l] + T[154, 155, k, l] * T[155, 136, k, l] + T[136, 135, k, l] * T[135, 154, k, l] + T[154, 135, k, l] * T[135, 136, k, l] + T[136, 155, k, l] * T[155, 156, k, l] + T[156, 155, k, l] * T[155, 136, k, l] + T[137, 136, k, l] * T[136, 155, k, l] + T[155, 136, k, l] * T[136, 137, k, l] + T[137, 138, k, l] * T[138, 156, k, l] + T[156, 138, k, l] * T[138, 137, k, l] + T[138, 156, k, l] * T[156, 155, k, l] + T[155, 156, k, l] * T[156, 138, k, l] + T[138, 156, k, l] * T[156, 157, k, l] + T[157, 156, k, l] * T[156, 138, k, l] + T[139, 138, k, l] * T[138, 156, k, l] + T[156, 138, k, l] * T[138, 139, k, l] + T[139, 140, k, l] * T[140, 157, k, l] + T[157, 140, k, l] * T[140, 139, k, l] + T[140, 157, k, l] * T[157, 156, k, l] + T[156, 157, k, l] * T[157, 140, k, l] + T[141, 140, k, l] * T[140, 157, k, l] + T[157, 140, k, l] * T[140, 141, k, l] + T[143, 142, k, l] * T[142, 168, k, l] + T[168, 142, k, l] * T[142, 143, k, l] + T[143, 169, k, l] * T[169, 207, k, l] + T[207, 169, k, l] * T[169, 143, k, l] + T[144, 145, k, l] * T[145, 187, k, l] + T[187, 145, k, l] * T[145, 144, k, l] + T[145, 146, k, l] * T[146, 187, k, l] + T[187, 146, k, l] * T[146, 145, k, l] + T[145, 187, k, l] * T[187, 146, k, l] + T[146, 187, k, l] * T[187, 145, k, l] + T[145, 187, k, l] * T[187, 209, k, l] + T[209, 187, k, l] * T[187, 145, k, l] + T[146, 145, k, l] * T[145, 187, k, l] + T[187, 145, k, l] * T[145, 146, k, l] + T[146, 187, k, l] * T[187, 276, k, l] + T[276, 187, k, l] * T[187, 146, k, l] + T[146, 147, k, l] * T[147, 276, k, l] + T[276, 147, k, l] * T[147, 146, k, l] + T[147, 276, k, l] * T[276, 187, k, l] + T[187, 276, k, l] * T[276, 147, k, l] + T[147, 148, k, l] * T[148, 189, k, l] + T[189, 148, k, l] * T[148, 147, k, l] + T[147, 276, k, l] * T[276, 188, k, l] + T[188, 276, k, l] * T[276, 147, k, l] + T[148, 189, k, l] * T[189, 188, k, l] + T[188, 189, k, l] * T[189, 148, k, l] + T[148, 147, k, l] * T[147, 276, k, l] + T[276, 147, k, l] * T[147, 148, k, l] + T[149, 152, k, l] * T[152, 153, k, l] + T[153, 152, k, l] * T[152, 149, k, l] + T[150, 192, k, l] * T[192, 212, k, l] + T[212, 192, k, l] * T[192, 150, k, l] + T[150, 149, k, l] * T[149, 279, k, l] + T[279, 149, k, l] * T[149, 150, k, l] + T[150, 149, k, l] * T[149, 151, k, l] + T[151, 149, k, l] * T[149, 150, k, l] + T[151, 149, k, l] * T[149, 152, k, l] + T[152, 149, k, l] * T[149, 151, k, l] + T[151, 193, k, l] * T[193, 194, k, l] + T[194, 193, k, l] * T[193, 151, k, l] + T[152, 153, k, l] * T[153, 194, k, l] + T[194, 153, k, l] * T[153, 152, k, l] + T[152, 149, k, l] * T[149, 279, k, l] + T[279, 149, k, l] * T[149, 152, k, l] + T[153, 194, k, l] * T[194, 193, k, l] + T[193, 194, k, l] * T[194, 153, k, l] + T[153, 194, k, l] * T[194, 196, k, l] + T[196, 194, k, l] * T[194, 153, k, l] + T[153, 154, k, l] * T[154, 171, k, l] + T[171, 154, k, l] * T[154, 153, k, l] + T[154, 153, k, l] * T[153, 194, k, l] + T[194, 153, k, l] * T[153, 154, k, l] + T[154, 171, k, l] * T[171, 172, k, l] + T[172, 171, k, l] * T[171, 154, k, l] + T[154, 155, k, l] * T[155, 172, k, l] + T[172, 155, k, l] * T[155, 154, k, l] + T[155, 154, k, l] * T[154, 171, k, l] + T[171, 154, k, l] * T[154, 155, k, l] + T[155, 172, k, l] * T[172, 171, k, l] + T[171, 172, k, l] * T[172, 155, k, l] + T[155, 172, k, l] * T[172, 173, k, l] + T[173, 172, k, l] * T[172, 155, k, l] + T[155, 156, k, l] * T[156, 173, k, l] + T[173, 156, k, l] * T[156, 155, k, l] + T[156, 155, k, l] * T[155, 172, k, l] + T[172, 155, k, l] * T[155, 156, k, l] + T[156, 173, k, l] * T[173, 172, k, l] + T[172, 173, k, l] * T[173, 156, k, l] + T[156, 173, k, l] * T[173, 174, k, l] + T[174, 173, k, l] * T[173, 156, k, l] + T[156, 157, k, l] * T[157, 174, k, l] + T[174, 157, k, l] * T[157, 156, k, l] + T[157, 174, k, l] * T[174,

173, k, l] + T[173, 174, k, l] * T[174, 157, k, l] + T[157, 156, k, l] * T[156, 173, k, l] + T[173, 156, k, l] * T[156, 157, k, l] + T[157, 174, k, l] * T[174, 175, k, l] + T[175, 174, k, l] * T[174, 157, k, l] + T[159, 175, k, l] * T[175, 176, k, l] + T[176, 175, k, l] * T[175, 159, k, l] + T[161, 176, k, l] * T[176, 175, k, l] + T[175, 176, k, l] * T[176, 161, k, l] + T[161, 176, k, l] * T[176, 177, k, l] + T[177, 176, k, l] * T[176, 161, k, l] + T[162, 177, k, l] * T[177, 176, k, l] + T[176, 177, k, l] * T[177, 162, k, l] + T[162, 177, k, l] * T[177, 178, k, l] + T[178, 177, k, l] * T[177, 162, k, l] + T[163, 178, k, l] * T[178, 177, k, l] + T[177, 178, k, l] * T[178, 163, k, l] + T[163, 178, k, l] * T[178, 179, k, l] + T[179, 178, k, l] * T[178, 163, k, l] + T[164, 179, k, l] * T[179, 178, k, l] + T[178, 179, k, l] * T[179, 164, k, l] + T[164, 179, k, l] * T[179, 180, k, l] + T[180, 179, k, l] * T[179, 164, k, l] + T[165, 180, k, l] * T[180, 179, k, l] + T[179, 180, k, l] * T[180, 165, k, l] + T[165, 180, k, l] * T[180, 181, k, l] + T[181, 180, k, l] * T[180, 165, k, l] + T[166, 181, k, l] * T[181, 180, k, l] + T[180, 181, k, l] * T[181, 166, k, l] + T[166, 181, k, l] * T[181, 182, k, l] + T[182, 181, k, l] * T[181, 166, k, l] + T[167, 182, k, l] * T[182, 181, k, l] + T[181, 182, k, l] * T[182, 167, k, l] + T[167, 182, k, l] * T[182, 183, k, l] + T[183, 182, k, l] * T[182, 167, k, l] + T[168, 183, k, l] * T[183, 182, k, l] + T[182, 183, k, l] * T[183, 168, k, l] + T[169, 170, k, l] * T[170, 208, k, l] + T[208, 170, k, l] * T[170, 169, k, l] + T[169, 207, k, l] * T[207, 237, k, l] + T[237, 207, k, l] * T[207, 169, k, l] + T[169, 207, k, l] * T[207, 208, k, l] + T[208, 207, k, l] * T[207, 169, k, l] + T[170, 169, k, l] * T[169, 207, k, l] + T[207, 169, k, l] * T[169, 170, k, l] + T[170, 208, k, l] * T[208, 207, k, l] + T[207, 208, k, l] * T[208, 170, k, l] + T[171, 197, k, l] * T[197, 196, k, l] + T[196, 197, k, l] * T[197, 171, k, l] + T[171, 197, k, l] * T[197, 198, k, l] + T[198, 197, k, l] * T[197, 171, k, l] + T[173, 174, k, l] * T[174, 198, k, l] + T[198, 174, k, l] * T[174, 173, k, l] + T[174, 198, k, l] * T[198, 197, k, l] + T[197, 198, k, l] * T[198, 174, k, l] + T[175, 174, k, l] * T[174, 198, k, l] + T[198, 174, k, l] * T[174, 175, k, l] + T[176, 177, k, l] * T[177, 202, k, l] + T[202, 177, k, l] * T[177, 176, k, l] + T[177, 202, k, l] * T[202, 204, k, l] + T[204, 202, k, l] * T[202, 177, k, l] + T[178, 177, k, l] * T[177, 202, k, l] + T[202, 177, k, l] * T[177, 178, k, l] + T[181, 182, k, l] * T[182, 184, k, l] + T[184, 182, k, l] * T[182, 181, k, l] + T[182, 183, k, l] * T[183, 185, k, l] + T[185, 183, k, l] * T[183, 182, k, l] + T[182, 184, k, l] * T[184, 185, k, l] + T[185, 184, k, l] * T[184, 182, k, l] + T[183, 182, k, l] * T[182, 184, k, l] + T[184, 182, k, l] * T[182, 183, k, l] + T[183, 185, k, l] * T[185, 184, k, l] + T[184, 185, k, l] * T[185, 183, k, l] + T[184, 185, k, l] * T[185, 186, k, l] + T[186, 185, k, l] * T[185, 184, k, l] + T[184, 206, k, l] * T[206, 204, k, l] + T[204, 206, k, l] * T[206, 184, k, l] + T[185, 184, k, l] * T[184, 206, k, l] + T[206, 184, k, l] * T[184, 185, k, l] + T[187, 209, k, l] * T[209, 210, k, l] + T[210, 209, k, l] * T[209, 187, k, l] + T[187, 276, k, l] * T[276, 210, k, l] + T[210, 276, k, l] * T[276, 187, k, l] + T[188, 276, k, l] * T[276, 210, k, l] + T[210, 276, k, l] * T[276, 188, k, l] + T[188, 189, k, l] * T[189, 211, k, l] + T[211, 189, k, l] * T[189, 188, k, l] + T[189, 188, k, l] * T[188, 276, k, l] + T[276, 188, k, l] * T[188, 189, k, l] + T[189, 148, k, l] * T[148, 279, k, l] + T[279, 148, k, l] * T[148, 189, k, l] + T[190, 191, k, l] * T[191, 192, k, l] + T[192, 191, k, l] * T[191, 190, k, l] + T[191, 192, k, l] * T[192, 212, k, l] + T[212, 192, k, l] * T[192, 191, k, l] + T[192, 212, k, l] * T[212, 213, k, l] + T[213, 212, k, l] * T[212, 192, k, l] + T[193, 194, k, l] * T[194, 195, k, l] + T[195, 194, k, l] * T[194, 193, k, l] + T[193, 213, k, l] * T[213, 212, k, l] + T[212, 213, k, l] * T[213, 193, k, l] + T[193, 213, k, l] * T[213, 214, k, l] + T[214, 213, k, l] * T[213, 193, k, l] + T[195, 214, k, l] * T[214, 213, k, l] + T[213, 214, k, l] * T[214, 195, k, l] + T[195, 214, k, l] * T[214, 215, k, l] + T[215, 214, k, l] * T[214, 195, k, l] + T[195, 194, k, l] * T[194, 196, k, l] + T[196, 194, k, l] * T[194, 195, k, l] + T[196, 216, k, l] * T[216, 215, k, l] + T[215, 216, k, l] * T[216, 196, k, l] + T[196, 216, k, l] * T[216, 217, k, l] + T[217, 216, k, l] * T[216, 196, k, l] + T[196, 197, k, l] * T[197, 217, k, l] + T[217, 197, k, l] * T[197, 196, k, l] + T[197, 196, k, l] * T[196, 216, k, l] + T[216, 196, k, l] * T[196, 197, k, l] + T[197, 217, k, l] * T[217, 216, k, l] + T[216, 217, k, l] * T[217, 197, k, l] + T[197, 217, k, l] * T[217, 218, k, l] + T[218, 217, k, l] * T[217, 197, k, l] + T[197, 198, k, l] * T[198, 199, k, l] + T[199, 198, k, l] * T[198, 197, k, l] + T[198, 197, k, l] * T[197, 217, k, l] + T[217, 197, k, l] * T[197, 198, k, l] + T[198, 199, k, l] * T[199, 200, k, l] + T[200, 199, k, l] * T[199, 198, k, l] + T[199, 200, k, l] * T[200, 201, k, l] + T[201, 200, k, l] * T[200, 199, k, l] + T[200, 201, k, l] * T[201, 218, k, l] + T[218, 201, k, l] * T[201, 200, k, l] + T[201, 218, k, l] * T[218,

217, k, l] + T[217, 218, k, l] * T[218, 201, k, l] + T[201, 218, k, l] * T[218, 219, k, l] + T[219, 218, k, l] * T[218, 201, k, l] + T[202, 204, k, l] * T[204, 205, k, l] + T[205, 204, k, l] * T[204, 202, k, l] + T[202, 204, k, l] * T[204, 206, k, l] + T[206, 204, k, l] * T[204, 202, k, l] + T[202, 203, k, l] * T[203, 205, k, l] + T[205, 203, k, l] * T[203, 202, k, l] + T[203, 202, k, l] * T[202, 204, k, l] + T[204, 202, k, l] * T[202, 203, k, l] + T[203, 205, k, l] * T[205, 204, k, l] + T[204, 205, k, l] * T[205, 203, k, l] + T[203, 219, k, l] * T[219, 218, k, l] + T[218, 219, k, l] * T[219, 203, k, l] + T[203, 219, k, l] * T[219, 220, k, l] + T[220, 219, k, l] * T[219, 203, k, l] + T[203, 205, k, l] * T[205, 220, k, l] + T[220, 205, k, l] * T[205, 203, k, l] + T[204, 206, k, l] * T[206, 252, k, l] + T[252, 206, k, l] * T[206, 204, k, l] + T[205, 204, k, l] * T[204, 206, k, l] + T[206, 204, k, l] * T[204, 205, k, l] + T[205, 203, k, l] * T[203, 219, k, l] + T[219, 203, k, l] * T[203, 205, k, l] + T[205, 220, k, l] * T[220, 219, k, l] + T[219, 220, k, l] * T[220, 205, k, l] + T[205, 220, k, l] * T[220, 221, k, l] + T[221, 220, k, l] * T[220, 205, k, l] + T[206, 252, k, l] * T[252, 251, k, l] + T[251, 252, k, l] * T[252, 206, k, l] + T[206, 252, k, l] * T[252, 255, k, l] + T[255, 252, k, l] * T[252, 206, k, l] + T[207, 237, k, l] * T[237, 236, k, l] + T[236, 237, k, l] * T[237, 207, k, l] + T[207, 236, k, l] * T[236, 237, k, l] + T[237, 236, k, l] * T[236, 207, k, l] + T[207, 208, k, l] * T[208, 255, k, l] + T[255, 208, k, l] * T[208, 207, k, l] + T[208, 207, k, l] * T[207, 236, k, l] + T[236, 207, k, l] * T[207, 208, k, l] + T[208, 255, k, l] * T[255, 253, k, l] + T[253, 255, k, l] * T[255, 208, k, l] + T[209, 187, k, l] * T[187, 276, k, l] + T[276, 187, k, l] * T[187, 209, k, l] + T[209, 210, k, l] * T[210, 276, k, l] + T[276, 210, k, l] * T[210, 209, k, l] + T[210, 209, k, l] * T[209, 238, k, l] + T[238, 209, k, l] * T[209, 210, k, l] + T[211, 230, k, l] * T[230, 222, k, l] + T[222, 230, k, l] * T[230, 211, k, l] + T[212, 213, k, l] * T[213, 222, k, l] + T[222, 213, k, l] * T[213, 212, k, l] + T[213, 222, k, l] * T[222, 223, k, l] + T[223, 222, k, l] * T[222, 213, k, l] + T[213, 214, k, l] * T[214, 231, k, l] + T[231, 214, k, l] * T[214, 213, k, l] + T[214, 213, k, l] * T[213, 222, k, l] + T[222, 213, k, l] * T[213, 214, k, l] + T[214, 231, k, l] * T[231, 230, k, l] + T[230, 231, k, l] * T[231, 214, k, l] + T[214, 195, k, l] * T[195, 215, k, l] + T[215, 195, k, l] * T[195, 214, k, l] + T[217, 218, k, l] * T[218, 224, k, l] + T[224, 218, k, l] * T[218, 217, k, l] + T[218, 224, k, l] * T[224, 225, k, l] + T[225, 224, k, l] * T[224, 218, k, l] + T[218, 219, k, l] * T[219, 226, k, l] + T[226, 219, k, l] * T[219, 218, k, l] + T[219, 218, k, l] * T[218, 224, k, l] + T[224, 218, k, l] * T[218, 219, k, l] + T[219, 226, k, l] * T[226, 227, k, l] + T[227, 226, k, l] * T[226, 219, k, l] + T[219, 220, k, l] * T[220, 227, k, l] + T[227, 220, k, l] * T[220, 219, k, l] + T[220, 227, k, l] * T[227, 226, k, l] + T[226, 227, k, l] * T[227, 220, k, l] + T[220, 219, k, l] * T[219, 226, k, l] + T[226, 219, k, l] * T[219, 220, k, l] + T[221, 220, k, l] * T[220, 227, k, l] + T[227, 220, k, l] * T[220, 221, k, l] + T[222, 230, k, l] * T[230, 231, k, l] + T[231, 230, k, l] * T[230, 222, k, l] + T[223, 222, k, l] * T[222, 230, k, l] + T[230, 222, k, l] * T[222, 223, k, l] + T[224, 225, k, l] * T[225, 281, k, l] + T[281, 225, k, l] * T[225, 224, k, l] + T[225, 281, k, l] * T[281, 232, k, l] + T[232, 281, k, l] * T[281, 225, k, l] + T[225, 228, k, l] * T[228, 232, k, l] + T[232, 228, k, l] * T[228, 225, k, l] + T[226, 227, k, l] * T[227, 229, k, l] + T[229, 227, k, l] * T[227, 226, k, l] + T[226, 228, k, l] * T[228, 229, k, l] + T[229, 228, k, l] * T[228, 226, k, l] + T[227, 229, k, l] * T[229, 228, k, l] + T[228, 229, k, l] * T[229, 227, k, l] + T[228, 229, k, l] * T[229, 233, k, l] + T[233, 229, k, l] * T[229, 228, k, l] + T[228, 232, k, l] * T[232, 233, k, l] + T[233, 232, k, l] * T[232, 228, k, l] + T[228, 225, k, l] * T[225, 281, k, l] + T[281, 225, k, l] * T[225, 228, k, l] + T[228, 232, k, l] * T[232, 281, k, l] + T[281, 232, k, l] * T[232, 228, k, l] + T[229, 228, k, l] * T[228, 232, k, l] + T[232, 228, k, l] * T[228, 229, k, l] + T[229, 233, k, l] * T[233, 232, k, l] + T[232, 233, k, l] * T[233, 229, k, l] + T[229, 233, k, l] * T[233, 234, k, l] + T[234, 233, k, l] * T[233, 229, k, l] + T[230, 231, k, l] * T[231, 239, k, l] + T[239, 231, k, l] * T[231, 230, k, l] + T[231, 239, k, l] * T[239, 240, k, l] + T[240, 239, k, l] * T[239, 231, k, l] + T[232, 281, k, l] * T[281, 235, k, l] + T[235, 281, k, l] * T[281, 232, k, l] + T[233, 232, k, l] * T[232, 235, k, l] + T[235, 232, k, l] * T[232, 233, k, l] + T[235, 251, k, l] * T[251, 250, k, l] + T[250, 251, k, l] * T[251, 235, k, l] + T[235, 251, k, l] * T[251, 252, k, l] + T[252, 251, k, l] * T[251, 235, k, l] + T[236, 280, k, l] * T[280, 254, k, l] + T[254, 280, k, l] * T[280, 236, k, l] + T[236, 207, k, l] * T[207, 237, k, l] + T[237, 207, k, l] * T[207, 236, k, l] + T[237, 236, k, l] * T[236, 280, k, l] + T[280, 236, k, l] * T[236, 237, k, l] + T[239, 243, k, l] * T[243, 244, k, l] + T[244, 243, k, l] * T[243, 239, k, l] + T[239, 243, k, l] * T[243,

```
242, k, l] + T[242, 243, k, l] * T[243, 239, k, l] + T[240, 239, k, l] * T[239, 243, k, l] + T[243, 239, k, l] *
T[239, 240, k, l] + T[240, 241, k, l] * T[241, 250, k, l] + T[250, 241, k, l] * T[241, 240, k, l] + T[242,
246, k, l] * T[246, 245, k, l] + T[245, 246, k, l] * T[246, 242, k, l] + T[242, 246, k, l] * T[246, 247, k, l] +
T[247, 246, k, l] * T[246, 242, k, l] + T[242, 247, k, l] * T[247, 248, k, l] + T[248, 247, k, l] * T[247,
242, k, l] + T[242, 243, k, l] * T[243, 248, k, l] + T[248, 243, k, l] * T[243, 242, k, l] + T[243, 248, k, l] *
T[248, 247, k, l] + T[247, 248, k, l] * T[248, 243, k, l] + T[243, 248, k, l] * T[248, 249, k, l] + T[249,
248, k, l] * T[248, 243, k, l] + T[244, 243, k, l] * T[243, 248, k, l] + T[248, 243, k, l] * T[243, 244, k, l] +
T[246, 242, k, l] * T[242, 247, k, l] + T[247, 242, k, l] * T[242, 246, k, l] + T[247, 248, k, l] * T[248,
263, k, l] + T[263, 248, k, l] * T[248, 247, k, l] + T[248, 263, k, l] * T[263, 262, k, l] + T[262, 263, k, l] *
T[263, 248, k, l] + T[249, 250, k, l] * T[250, 256, k, l] + T[256, 250, k, l] * T[250, 249, k, l] + T[250,
251, k, l] * T[251, 259, k, l] + T[259, 251, k, l] * T[251, 250, k, l] + T[250, 256, k, l] * T[256, 257, k, l] +
T[257, 256, k, l] * T[256, 250, k, l] + T[251, 259, k, l] * T[259, 260, k, l] + T[260, 259, k, l] * T[259,
251, k, l] + T[251, 259, k, l] * T[259, 258, k, l] + T[258, 259, k, l] * T[259, 251, k, l] + T[252, 251, k, l] *
T[251, 259, k, l] + T[259, 251, k, l] * T[251, 252, k, l] + T[252, 255, k, l] * T[255, 253, k, l] + T[253,
255, k, l] * T[255, 252, k, l] + T[253, 254, k, l] * T[254, 280, k, l] + T[280, 254, k, l] * T[254, 253, k, l] +
T[253, 280, k, l] * T[280, 254, k, l] + T[254, 280, k, l] * T[280, 253, k, l] + T[254, 253, k, l] * T[253,
255, k, l] + T[255, 253, k, l] * T[253, 254, k, l] + T[256, 264, k, l] * T[264, 265, k, l] + T[265, 264, k, l] *
T[264, 256, k, l] + T[257, 256, k, l] * T[256, 264, k, l] + T[264, 256, k, l] * T[256, 257, k, l] + T[260,
277, k, l] * T[277, 266, k, l] + T[266, 277, k, l] * T[277, 260, k, l] + T[260, 277, k, l] * T[277, 269, k, l] +
T[269, 277, k, l] * T[277, 260, k, l] + T[262, 263, k, l] * T[263, 267, k, l] + T[267, 263, k, l] * T[263,
262, k, l] + T[264, 265, k, l] * T[265, 268, k, l] + T[268, 265, k, l] * T[265, 264, k, l] + T[266, 277, k, l] *
T[277, 269, k, l] + T[269, 277, k, l] * T[277, 266, k, l] + T[266, 265, k, l] * T[265, 268, k, l] + T[268,
265, k, l] * T[265, 266, k, l]);     #Objective function

subj to LaneConstraint {(i, j) in ARCS : (i, j) not in GRADERARCS}:     #One of these constraints for
each arc
        sum {k in POSITIONS, l in PLOWS} T[i, j, k, l] >= PASSES[i, j];          #At least as many
passes

subj to GraderConstraint {(i, j) in GRADERARCS : (i, j) in ARCS}:       #One of these constraints for
each grader arc
        sum {k in POSITIONS} (T[i, j, k, 1] + T[i, j, k, 2]) >= PASSES[i, j];       #At least as many
passes

subj to OneArcPerPosition {k in POSITIONS, l in PLOWS}:      #Only one arc in each position of each
route
        sum {(i, j) in ARCS} T[i, j, k, l] <= 1;

subj to ConnectedArcs {j in J, k in POSITIONS, l in PLOWS : k != p}:
        (sum {(j, g) in ARCS} T[j, g, (k + 1), l]) - (sum {(i, j) in ARCS} T[i, j, k, l]) <= 0;

subj to Depot {l in PLOWS}:     #Arc from node 270 is always in the first position of each route
        sum {(270, j) in ARCS} T[270, j, 1, l] = 1;
```

Appendix L - Source Code Used to Decode AMPL Output

```
import re

#-----------------------------------------------------------------------------------------------------------------
#Function definitions
#-----------------------------------------------------------------------------------------------------------------

def FileConversionSpaceDelimited(Path):
    ReturnList = []
    DataFile = open(Path, "r")
    for Line in DataFile:
        #print(Line)
        LineString = str(Line)
        LineList = LineString.split()
        ReturnList.append(LineList)
    DataFile.close()
    return ReturnList

def FileConversion(Path):
    ReturnList = []
    DataFile = open(Path, "r")
    for Line in DataFile:
        LineList = RemoveFirstItem(eval(Line))
        ReturnList.append(LineList)
    DataFile.close()
    return ReturnList

def ExtractIndexes(FileList):
    IndexList = []
    for Line in FileList:
        if len(Line) > 0:
            if (Line[0] == 'T') or (Line[0][0] == '['):
                KLIndexes = [(FileList.index(Line) + 2), (FileList.index(Line) + 282)]
                IndexList.append(KLIndexes)
                # MyString = str(Line) + str(FileList.index(Line))
                # print(MyString)
    return IndexList

def WriteOutputFile(List, File):          #Writes the contents of a list to a file
    OutputFile = open(File, "w")          #Opens the file
    for Item in List:          #Loops through the list
```

```
        OutputFile.write(str(Item) + "\n")          #Writes each item to the file
    OutputFile.close()          #Close the file
    return True         #Returns true to indicate success

def CleanStringForFileName(MyString):
    ReturnString = ''
    for Character in MyString:
        if Character != '*':
            ReturnString += Character
    return ReturnString

def CreateMatrix(n):
    ReturnList = []
    for x in range(n):
        ReturnList.append([])
        for y in range(n):
            ReturnList[x].append(0)
    return ReturnList

def PrintList(List):
    for Item in List:
        print(Item)

def RemoveFirstItem(List):
    ReturnList = []
    FirstItem = True
    for Item in List:
        if FirstItem:
            FirstItem = False
        else:
            ReturnList.append(Item)
    return ReturnList

#--------------------------------------------------------------------------------------------------------------------
#Main
#--------------------------------------------------------------------------------------------------------------------

FileList = FileConversionSpaceDelimited("large_results.txt")

IndexList = ExtractIndexes(FileList)

ListOfFileNames = []
```

```
for IndexPair in IndexList:
    KLList = []
    for n in range(IndexPair[0], (IndexPair[1] + 1)):
        KLList.append(FileList[n])
    KL = str(FileList[(IndexPair[0] - 2)])
    OutputFileName = CleanStringForFileName(KL) + ".txt"
    ListOfFileNames.append(OutputFileName)
    WriteOutputFile(KLList, OutputFileName)

#print(ListOfFileNames)

FileNamesSortedByPlow = [[],[],[],[],[],[],[],[],[]]

for FileName in ListOfFileNames:
    FileNamesSortedByPlow[int(FileName[len(FileName) - 8])-1].append(FileName)

for Plow in FileNamesSortedByPlow:
    PlowList = CreateMatrix(281)
    for FileName in Plow:
        KLFileList = FileConversion(FileName)
        for Row in KLFileList:
            i = KLFileList.index(Row)
            for Item in Row:
                j = Row.index(Item)
                if Item == '1':
                    PlowList[i][j] = 1
    ArcsServicedList = []
    for Row in PlowList:
        for Item in Row:
            if Item == 1:
                ArcsServicedList.append([PlowList.index(Row)+1, Row.index(Item)+1])
    PlowString = "Plow " + str(FileNamesSortedByPlow.index(Plow)+1) + " services arcs " +
str(ArcsServicedList)
    print(PlowString)
```

Appendix M - Map Displaying Optimized Route Recommendations