CSE2246 Assignment 2

Traveling Salesmen Problem with Penalty (TSPwP)

Due date: June 4th, 2025

The traveling salesman problem (TSP) asks the following question: Given a list of cities and the distances between each pair of cities, what is the shortest possible route that visits each city exactly once and returns to the origin city? In this assignment, we will focus on a variant of TSP, TSP with penalty (TSPwP). In this variant, each point carries a penalty if not visited. Your goal is to minimize tour length plus penalties for skipped points.

Inputs are: n cities, with their locations (x and y coordinates) in a 2D grid, and a penalty value.

Output is: Ordering (tour) of the cities traveled by the salesman, which includes a subset of the cities. Also the total tour length plus penalties should be given as the output.

Distance between two cities is defined as the Euclidian distance rounded to the nearest integer. In other words, you will compute distance between two cities $c_1 = (x_1, y_1)$ and $c_2 = (x_2, y_2)$ as follows:

$$d(c_1, c_2) = round\left(\sqrt{(x_1 - x_2)^2 + (y_1 - y_2)^2}\right)$$

For example, if three cities are given with the coordinates $c_1 = (0,2), c_2 = (2,3)$ and $c_3 = (3,0)$, then a tour with ordering $c_1,\ c_2,\ c_3,\ c_1$ has a total distance of :

$$rnd(\sqrt{5}) + rnd(\sqrt{10}) + rnd(\sqrt{13}) = rnd(2{,}23) + rnd(3{,}16) + rnd(3{,}60) = 2 + 3 + 4 = 9$$

Project Specification:

Your team (a group of three students) is asked to design and implement a method for finding a solution for the problem that is as close to the optimal solution as possible. Note that the TSP problem is an NP-hard problem, and TSPwP is at least as hard as the ordinary TSP problem. Given that you must determine the initial city, which cities to exclude and the order of cities to visit, achieving optimal solutions proves exceedingly challenging. Your goal is not to design an algorithm for the optimal solution, rather you are requested to do your best. This is an open-ended project.

You may do the following:

- Read as much as you want to learn about how to solve the problem. But **you have to cite** any resources you use. You may want to start with some approximation algorithms (such as local search heuristics) in Chapter 12.
- You may use whichever programming language you want.

You may **not use** the following:

- Existing implementations or subroutines
- Extensive libraries (if you are not sure, check with the instructor)
- Other people's code.

Input format: The input will always be provided as a text file. The first line specifies the fixed penalty for not visiting a city, which applies equally to all cities. Each subsequent line describes a city using three space-separated values: the city ID, the x-coordinate, and the y-coordinate. The final line of the input file is blank.

Output format: You must output your solution into another text file. The first line should contain two values: the total length of the tour plus penalties, and the number of cities visited, denoted as n_v. The next n_v lines must list the city IDs in the order they are visited in the tour. The final line of the output file should be left blank.

Important note: We will provide some sample inputs, and a verifying procedure in Google classroom. This verifying procedure will verify that your output is a VALID output for a given input. (It will not verify the optimality of the code, but only its validity). Using the verifier, you should test that your program reports the correct total distance. Also, note that your code should work in a reasonable time for the inputs with up to 50.000 cities.

Test instances: On June 3rd, we will announce 4 test instances on the website. By June 4th, 23:59, you will be required to submit 4 separate output text files (according to the output format) corresponding to each of these test instances**.**

Project report: Together with the output text files, you should also submit a project report. The project report should describe the ideas behind your algorithm as completely as possible. It should not exceed 3 pages in length in no less than 10pt. **You should also describe the "division of labor – who did what?".**

Grading policy:

60% of your grade will be determined by your project report. Clarity and creativity of your work will significantly affect your grade.

40% of your grade will be determined by your solutions to the test instances. Studies that find better solutions will get higher grades.

Come on to the contest: "Hodri Meydan!"

For each of the four test inputs, we will identify the best, the second-best, and the third-best performing teams. Best, second, and third projects receive 3 stars, 2 stars, and 1 star, respectively. So if a project performs best in all four test inputs it will receive 3*4=12 stars.

Stars	1	2	3	4	5	6	7	8	9	10	11	12
Bonus Points	5	7	9	11	13	15	17	19	21	23	25	30