

Optimization Methods - Assignment 1.1

A Description

The Traveling Salesman Problem (TSP) is a classic optimization problem in which a salesman is required to visit a set of cities exactly once and return to the starting city. The objective is to find the shortest possible route that visits all cities. The TSP is a well-known NP-hard problem, and it has applications in various fields, including logistics, transportation, and supply chain management.

The objective of this assignment is to implement a C++ program using IBM CPLEX to solve the Traveling Salesman Problem (TSP). Students are required to:

- Build a mathematical model of the TSP using IBM CPLEX in C++.
- Justify the choice of the subtour elimination constraints.
- Solve the TSP model using CPLEX.
- Validate the solution obtained using at least 20 TSP instances of different sizes.
- Deliver a comprehensive report.
- Submit the source code of the implemented program.

B Resources

The following resources are provided to assist students in completing the assignment:

- TSP instances in the folder *assets/data/*.
- Sample code snippets for reading data files and building the TSP model in the folder *assets/tspcode/*.
- IBM CPLEX documentation and tutorials.
- Assignment guidelines and requirements in *NEXXUS* platform.

C Submission Guidelines

Students are required to submit the following deliverables by the specified deadline:

- A comprehensive report in PDF format detailing the implementation, solution process, and results.
- A zip file containing the source code of the implemented program.

The report must include the following sections:

1. Introduction: Briefly describe the Traveling Salesman Problem and its significance.
2. Mathematical Model: Present the mathematical model of the TSP using IBM CPLEX.
3. Implementation: Describe the implementation in C++ using IBM CPLEX.
4. Solution Process: Explain the process of solving the TSP model and validating the solution.
5. Results: Present the results obtained using tables, graphs, and visualizations.
6. Conclusion: Summarize the key findings and insights from the assignment.
7. References: Include any references used in the assignment.

Note that the report should be well-structured, clearly written, and properly formatted. The source code should be well-documented and organized for easy understanding. Late submissions will not be accepted.