COT6936 – Project 2

Objective

Students will be able to design, implement, test, and compare solutions for known combinatorial optimization problems using concepts and tools studied in class.

Assignment Problem

Design and implement a comparative study to study the performance of the methods of local search and genetic algorithms applied to the following problem. Given a set of n points

$$(x_0, y_0),\ (x_1, y_1),\ \ldots,\ (x_{n-1}, y_{n-1})$$

in the plane, compute the shortest path that joins all of the points and starts and ends in the same point. This is a variation of the traveling salesman problem, in which the graph vertices are represented by the points in the plane and the weights on the edges are represented by the pairwise distances.

Requirements

- The graph will be represented with an adjacency matrix, with distances as weights
- Input point coordinates will be integers, and weights will be positive real numbers
- The information of the data points will be provided in a text file
- A *local search* algorithm and a *genetic algorithm* will be used in providing a solution to the problem
- Output will consist of the sequence of points obtained by each method, as well as the total distance in each case, printed to the console.
- A .csv file will be created by your application where the best distance at each iteration of each method is saved.
- Plot, using Excel, the content of the .csv file. 2-D charts are recommended.
- The number of input points should be, at least, 50.

Please consult the class materials for a discussion on local search and genetic algorithms.

Input file format

The input file will have this format:

- number of points
- list of the points, one point per line

For example,

```
6                   data set has 6 points
100 62
120 55
38 70          (x, y) points
60 220
345 420
10 317
```

Extra Mile

Bonus points will be granted if, in addition to the solution of the problem described above, the following is completed:

1) your project considers a third heuristic in the study, simulated annealing,
2) output of the set of points and best route as obtained by each method is graphical.

Guidelines

- The assignment is to be completed in teams of up-to three students.
- The given problem is based on the content studied in class on combinatorial optimization algorithms.
- You are allowed to use the code discussed in the lectures or in class. In those cases, make sure you properly credit its source.
- No installation of any software tool, package, or code should be required by anyone interested in running your code, other than a Java IDE and the Java™ Platform, Standard Edition Development Kit.

Deliverables:

- A compressed folder, *names_Project 2* (e.g. *Smith_John_Project 2*), containing:
  1. all of the source code of the exercise
  2. Conclusions (Word or PDF file): a document explaining what you observed in the experiment and your conclusions. This document will include the text of your explanations and the picture(s), chart(s), or diagram(s) obtained in Excel
  3. Screenshots of the running program (screenshots are *images*; they include, at least, part of the IDE and the code, and the complete graphical output). Screenshots will correspond to the solution of **two** different point sets, each with a minimum of 50 points, and are to be embedded in the Conclusions document.
  4. The two text files containing the set of points used for the outputs in the screenshots
  5. The .csv file and the corresponding Excel file
- Include **only** the .java files (source code); do not include other files or folders generated by the IDE.
- Make sure you write name(s) and PID(s) in the class comment section of each Java file, in the Conclusions document, as well as in the comment window of the project dropbox.

- Only one member of a team will submit the project.

Grading Rubric

The assignment is worth 20 points (out of 100 total course points). Grade components:

<table>
<tr><th>Component</th><th>Points</th><th>Description</th></tr>
<tr><td>Submission</td><td>1</td><td>The student has submitted the project solution using the requirements for deliverables specified in the <i>Deliverables</i> section.</td></tr>
<tr><td>Organization</td><td>1</td><td>Code is expected to be neat, organized, and readable.</td></tr>
<tr><td>Content</td><td>18</td><td><table>
<tr><th>Deliverable</th><th>Points</th></tr>
<tr><td>source code</td><td>12 pts</td></tr>
<tr><td>Input files</td><td>1 pts</td></tr>
<tr><td>conclusions</td><td>3 pts</td></tr>
<tr><td>.csv file</td><td>1 pts</td></tr>
<tr><td>Excel file</td><td>1 pts</td></tr>
</table></td></tr>
<tr><td rowspan="2">Extra Mile (*)</td><td>3</td><td>1) Inclusion of simulated annealing in the study</td></tr>
<tr><td>1</td><td>2) Graphical output</td></tr>
</table>

(*) Course extra credit. Parts 1) and 2) are independent, i.e., you can implement one, the other, or both. Note that for each of 1) and 2) points will be granted only if solution is complete; partial work will not be considered for a grade.