Game Programming Using C++ - Final Project - FALL 2018

Write a program that solves the travelling salesman problem and visualizes the solution.

**Requirements**

1- You must use simulated annealing to solve the problem.

2- You must create function “retrieve\_Nodes” that populates the nodes.

This function must call “create\_Random\_Nodes” which populates the nodes with random positions.

Note: “create\_Random\_Nodes” inside “retrieve\_Nodes” can be replaced in the future to allow acquiring nodes from a file or internet.

3- In order to solve the problem, you must provide a list of nodes, a beginning node and an ending node index to the algorithm.

4- You must visualize nodes and the generated path (the current solution) in every iteration of annealing.

The beginning and ending nodes must be indicated visually.

5- Once temperature is below a certain threshold, you must reset the simulation to start a new one using “retrieve\_Nodes”. Your simulation must loop indefinitely (watch out for memory leaks!)

6- A simple project template can be found here <https://github.com/leventalbayrak/Levent_Development/tree/master/game_programming_using_C%2B%2B_final/SDL_solution>

**Grading (no partial points for items)**

1-Create “retrieve\_Nodes” and “create\_Random\_Nodes” functionality, and use them appropriately – 20pts

2-Generate nodes and visualize (including edges/paths between nodes) a solution – 10pts

3-Working simulated annealing– 40pts

4-Simulation resets correctly without crashing – 10pts

5-Functional github link/visual studio solution – 10pts (see below)

You must add user leventalbayrak as a collaborator – 5pts

The visual studio solution must work correctly and your program must run – 5pts

6-A gif of at least 2 cycles (resets) of your simulation inside your project folder – 10pts

Bonus points (50 pts): Correctly utilize priority queues for distance calculations.

**Submission**

Submit to Canvas:

1- The github repository link to your solution

The repository must contain:

a- A gif (you may use an external program such as gyazo) of at least 2 cycles (resets) of your simulation.

b- Your solution file which is ready to run!

2- The collaboration invitation link