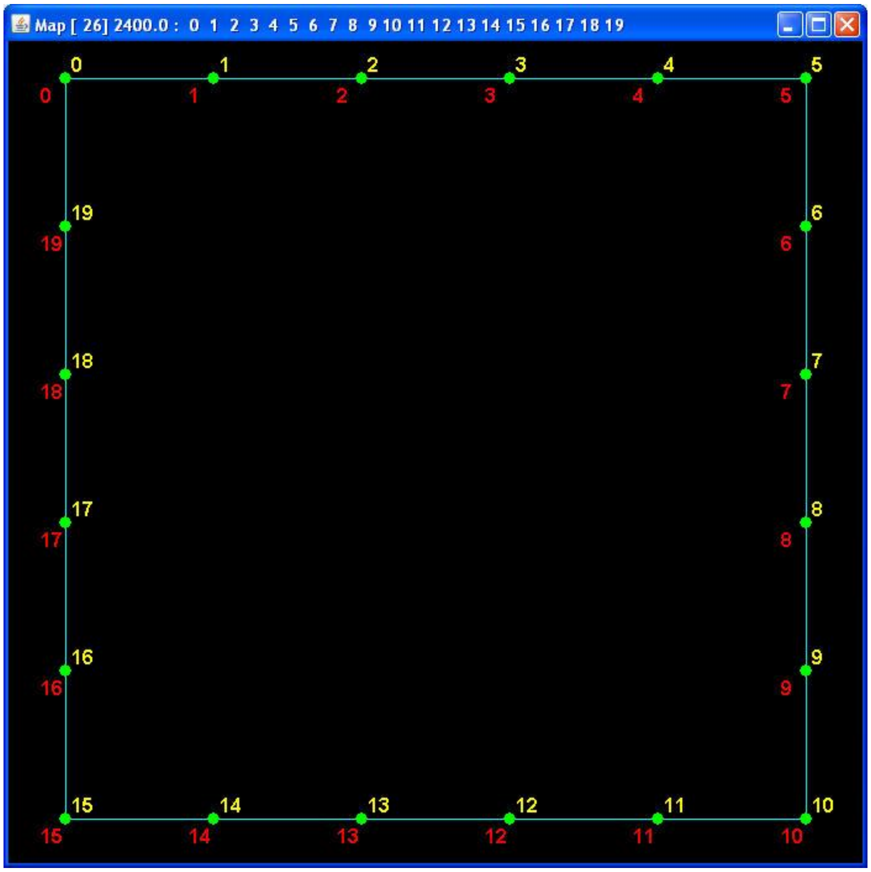
Assignment: A genetic algorithm (GA) for solving instances of the Traveling Salesman Problem (TSP).

In this assignment you will implement a GA solution to the Traveling Salesman Problem. In the figure below you have 20 cities (numbered in yellow from 0 -- 19) laid out on the sides of a square. Your program will start off with an initial collection of random tours. Your GA will determine the optimal way to traverse these 20 cities which is along the sides of the square (below right figure). The square is 500 units wide so the optimal tour distance is 2000.



Map [26] 2400.0 : 0 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19

Image from [*http://nifty.stanford.edu/2008/sooriamurthi-ga-tsp/GA-TSP2.jpg*](http://nifty.stanford.edu/2008/sooriamurthi-ga-tsp/GA-TSP2.jpg)

General procedure

You can a concise description of the essential ideas behind a genetic algorithm read the chapter from A.K.Dewdney's *The New Turing Omnibus*. The essential procedure is:

1. Determine a way to *encode* a representation for a solution.
2. Determine a *fitness function* (how to compare two candidate solutions).

3. Create an initial population of random individuals.
4. REPEAT
    1. **SELECT** two parents from the population pool
    2. **CROSSOVER** (reproduce) the parents to produce two children
    3. probabilistically **MUTATE** the children
    4. collect the children in a new pool

UNTIL
the fitness of the best individual doesn't improve (or the average fitness of the pool as a whole doesn't improve)

You are free (advised, even) to use an AI chatbot to generate an initial solution. Add comments to each function explaining what it does.

Submission
Submit to Canvas your code and a sample execution of your program.