Assignment Two

SOFE 3720/ CSCI 4610: Introduction to Artificial Intelligence/ Artificial Intelligence

Winter 2019

Dr. Sukhwant Kaur Sagar

Submission Deadline: Tuesday, March 20, 2019, 11:59 PM

Submission Guidelines

We are assuming the same assignment groups as the assignment1 groups. No new groups are allowed.

Student Number	Banner id	Student Name
1.		
2.		
3.		

Note:

a. **Email submissions will not be accepted.**
b. **Only one copy of the solution to this problem needs to be submitted per team.**
c. **Make sure that all of you contribute equally; in your write up, explain how you divided the work.**

Implementation of Genetic Algorithm for finding solution of Travelling Salesman Problem

A genetic algorithm (GA) is great for finding solutions to complex search problems. They're often used in fields such as engineering to create incredibly high quality products .Thanks to their ability to search a through a huge combination of parameters to find the best match.

The basic process for a genetic algorithm is:

1. Initialization - Create an initial population. This population is usually randomly generated and can be any desired size, from only a few individuals to thousands.

2. Evaluation - Each member of the population is then evaluated and we calculate a 'fitness' for that individual. The fitness value is calculated by how well it fits with our desired requirements. These requirements could be simple, 'faster algorithms are better', or more complex, 'stronger materials are better but they shouldn't be too heavy'.

3. Selection - We want to be constantly improving our populations overall fitness. Selection helps us to do this by discarding the bad designs and only keeping the best individuals in the population. There are a few different selection methods but the basic idea is the same, make it more likely that fitter individuals will be selected for our next generation.

4. Crossover - During crossover we create new individuals by combining aspects of our selected individuals. We can think of this as mimicking how sex works in nature. The hope is that by combining certain traits from two or more individuals we will create an even 'fitter' offspring which will inherit the best traits from each of its parents.

5. Mutation - We need to add a little bit randomness into our populations' genetics otherwise every combination of solutions we can create would be in our initial population. Mutation typically works by making very small changes at random to an individual’s genome.

6. And repeat! - Now we have our next generation we can start again from step two until we reach a termination condition.

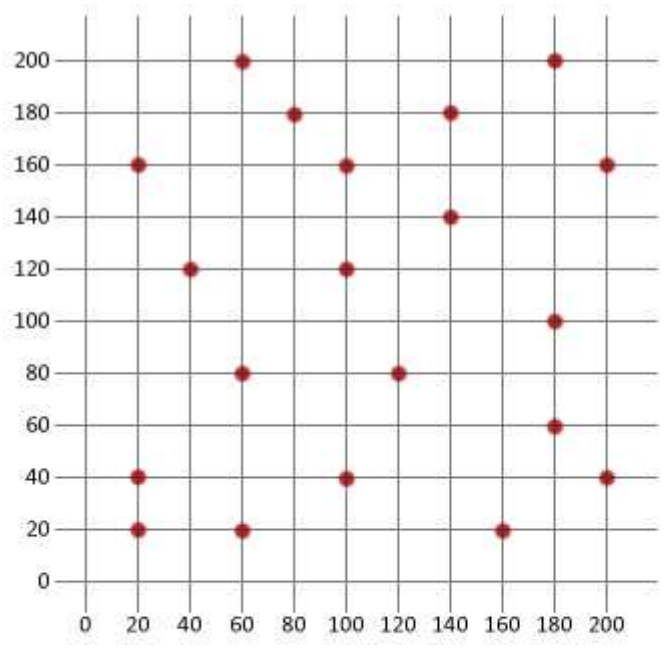
Termination

There are a few reasons why you would want to terminate your genetic algorithm from continuing its search for a solution. The most likely reason is that your algorithm has found a solution which is good enough and meets a predefined minimum criteria. Offer reasons for terminating could be constraints such as time or money.

<u>Problem Description:</u>

Imagine you're a salesman and you've been given a map as shown. On it you see that the map contains a total of 20 locations and you're told it's your job to visit each of these locations to make a sell. Before you set off on your journey you'll probably want to plan a route so you can minimize your travel time.

Fortunately, humans are pretty good at this, we can easily work out a reasonably good route without needing to do much more than glance at the map. **However, when we've found a route that we believe is optimal, how can we test if it's really the optimal route?**



Well, in short, we can't - at least not practically.

To understand why it's so difficult to prove the optimal route let's consider a similar map with just 3 locations instead of the original 20. To find a single route, we first have to choose a starting location from the three possible locations on the map. Next, we'd have a choice of 2 cities for the second location, then finally there is just 1 city left to pick to complete our route. This would mean there are 3 x 2 x 1 different routes to pick in total.

That means, for this example, there are only 6 different routes to pick from. So for this For example, the factorial of 10 is 3628800, but the factorial of 20 is a gigantic, 2432902008176640000.

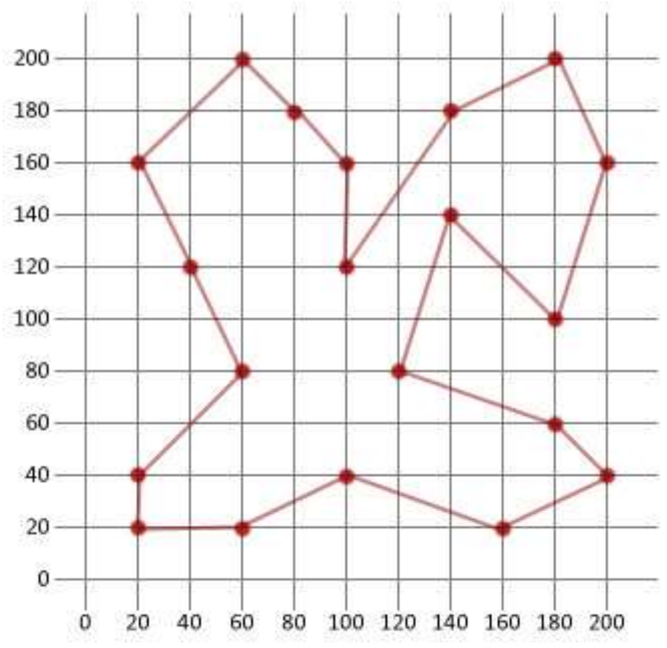
So going back to our original problem, if we want to find the shortest route for our map of 20 locations we would have to evaluate 2432902008176640000 different routes! Even with modern computing power this is terribly impractical, and for even bigger problems, it's close to impossible.

You are required to build up a solution to this TSP problem by using Genetic Algorithm

Solution: Finding a solution to the travelling salesman problem requires we set up a genetic algorithm in a specialized way. For instance, a valid solution would need to represent a route where every location is included at least once and only once. If a route contain a single location more than once, or missed a location out completely it wouldn't be valid and we would be valuable computation time calculating its distance.

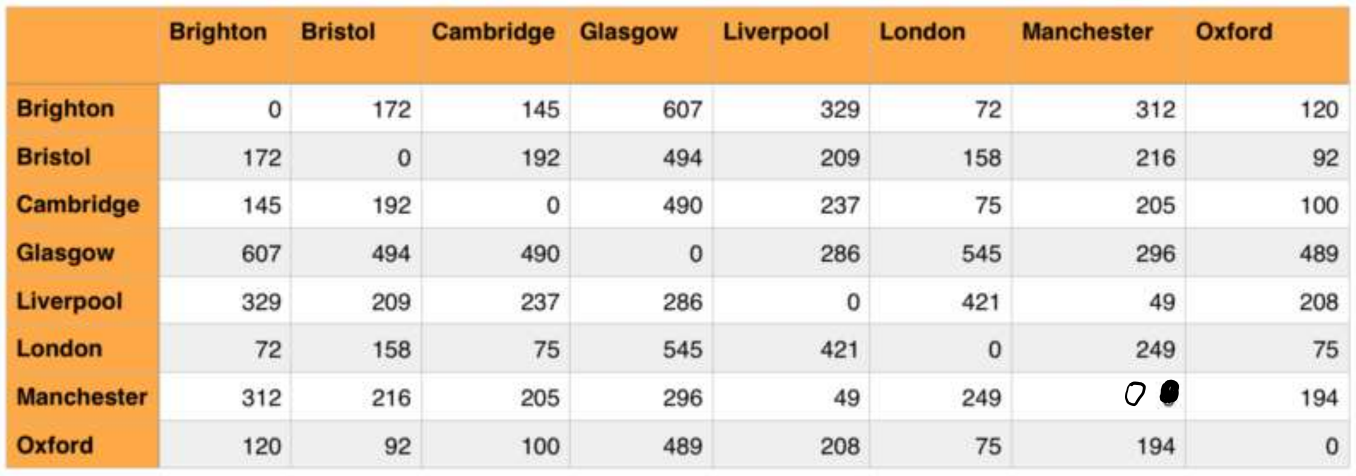
To ensure the genetic algorithm does indeed meet this requirement special types of mutation and crossover methods might be needed.

Final Results: It should be shown graphically (let's say) like this:



and the total cost and the cities involved.

Test your algorithm for the following problem domain too, and attach the results in the document. The table below shows the distances between each city in kilometres.



	Brighton	Bristol	Cambridge	Glasgow	Liverpool	London	Manchester	Oxford
Brighton	0	172	145	607	329	72	312	120
Bristol	172	0	192	494	209	158	216	92
Cambridge	145	192	0	490	237	75	205	100
Glasgow	607	494	490	0	286	545	296	489
Liverpool	329	209	237	286	0	421	49	208
London	72	158	75	545	421	0	249	75
Manchester	312	216	205	296	49	249	0	194
Oxford	120	92	100	489	208	75	194	0

Problem Domain 2

Submission/grading

1. You will submit your code (both in report as well as link to GitHub repository) - 40%
2. A write-up showing some graphical outputs. The write-up will contain discussion of all 5 components of Genetic Algorithm about how they were calculated and what assumptions were taken for both the problem domains. - 30%
3. You need to clarify the termination condition- 10%
4. Comparison of the results in tabular format when different methods of crossover and mutation are used- 20 %

Supporting Documents for your reference is also available on BB. However, you may have to research more on solving the given problem. You may use any programming language to implement the Genetic Algorithm

***GOOD LUCK**