This is a Summary of a Project by Ahmad Najiy Wahab

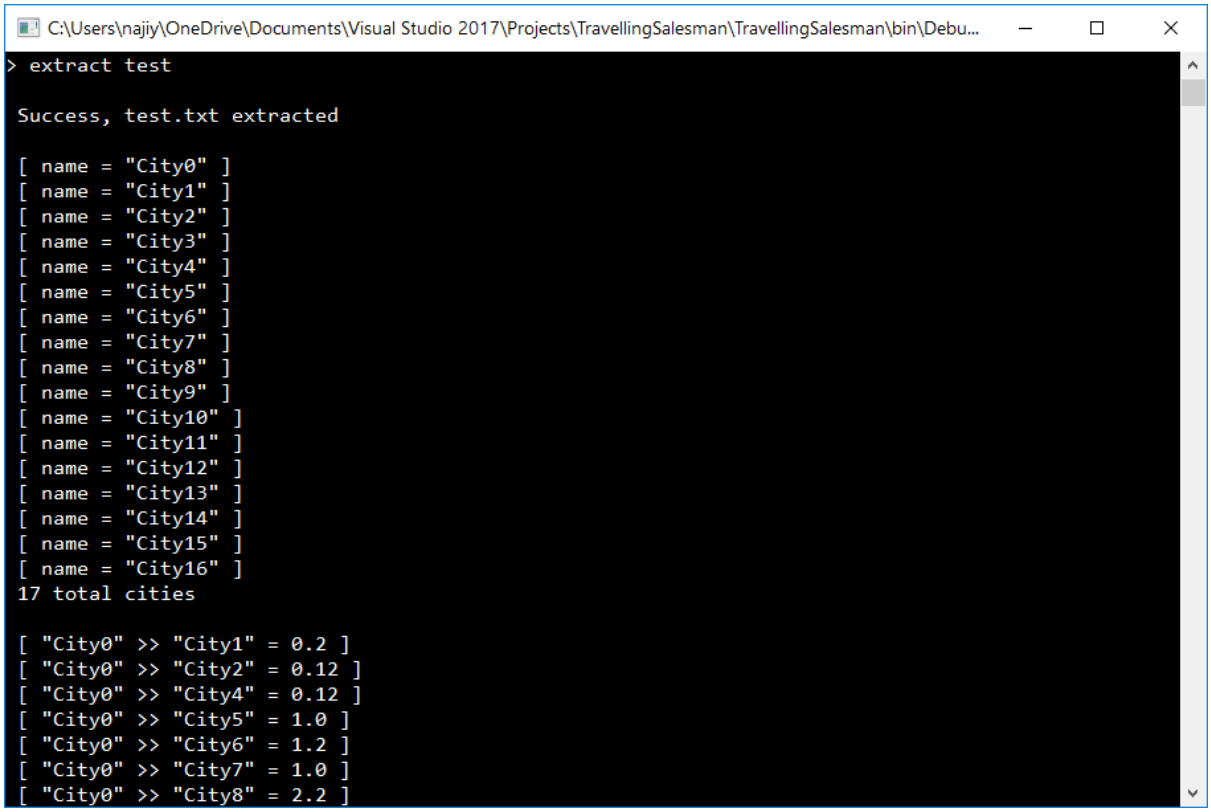
Emails: najiy.wa@googlemail.com **& ahmad.wahab@mail.bcu.ac.uk**

Contents

Travelling Salesman Problem with Genetic Algorithm

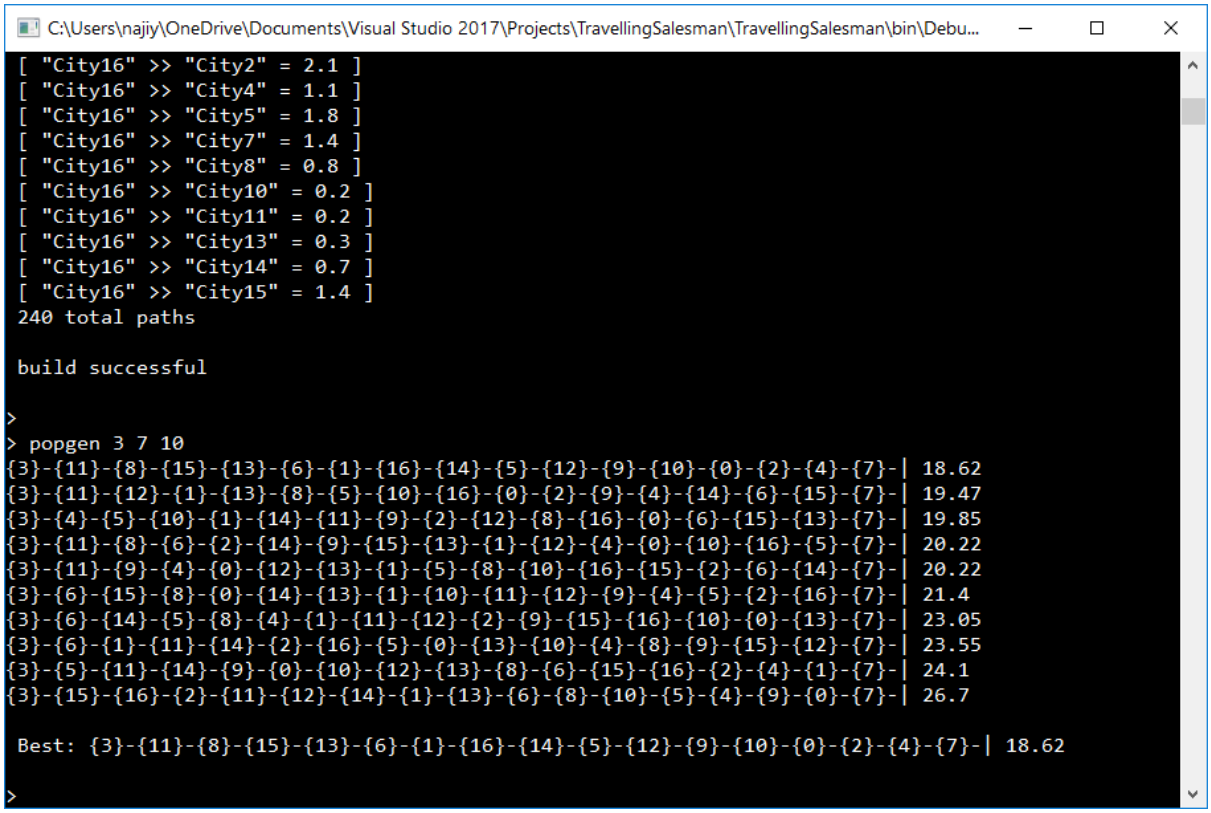
Extracting Arrays From File

Before the program can use GA commands, the cities need to be extracted using the command ‘extract’ followed by the filename (expected file is filename.txt – give name only). The command extract, takes the array in the file and creates the City-network structure displaying the result of cities added and paths added.



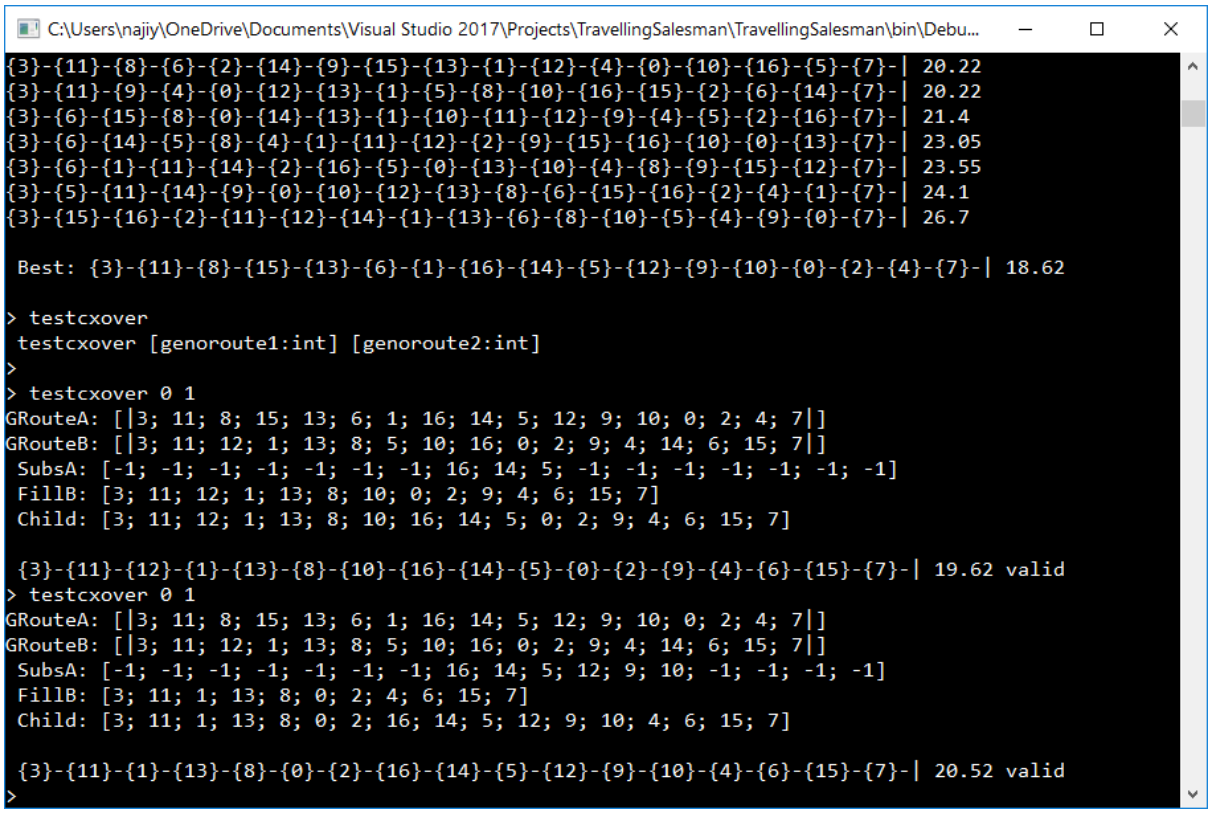
Populating the Initial Generation

To initialise the population, the command ‘popgen 3 7 10’ is used to give a population of ten with members from fourth city (start) and eighth city (end). The members are ranked shortest first by distance.



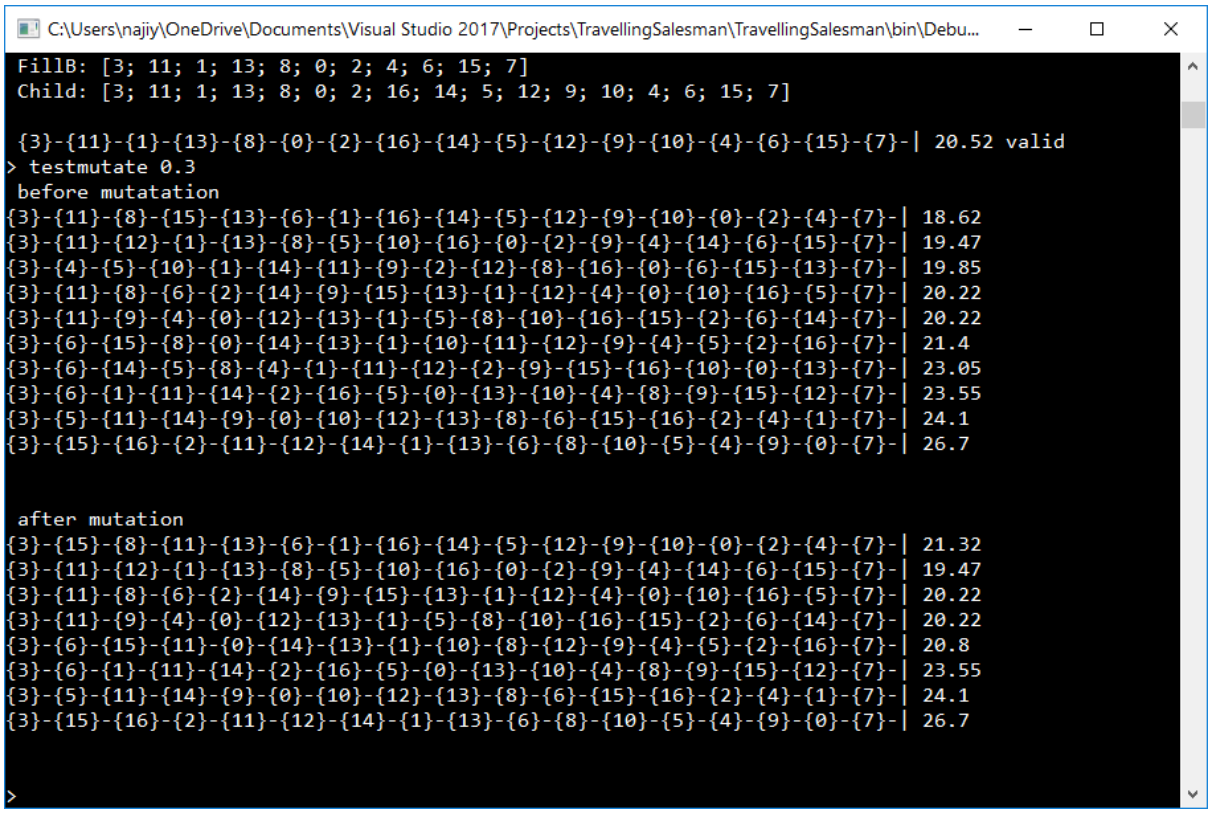
Crossover Test Process

Below shows the cross-over test result using command ‘testcxover 0 1’. First and second arguments are the parents from the genepool (argument is parent position in genepool). The result is successful in that the cross-over takes a subsection of the first then fill the rest with the second (in order of). Cross-over test below:



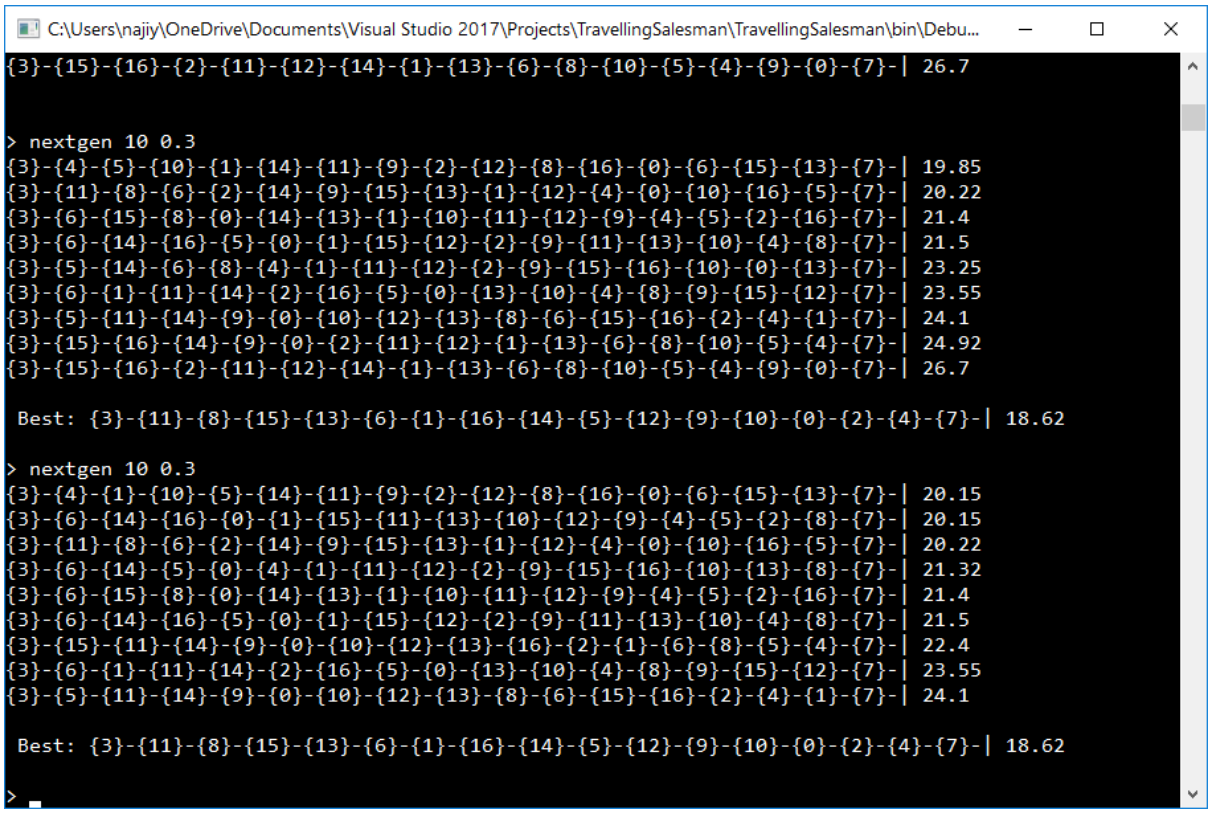
Mutation Test Process

The test mutation below with thirty percent probability shows before the genepool undergo mutation process and after the genepool having gone through mutation process. Second, third and second last members are mutated. Invalid mutations are removed from the genepool as show below:

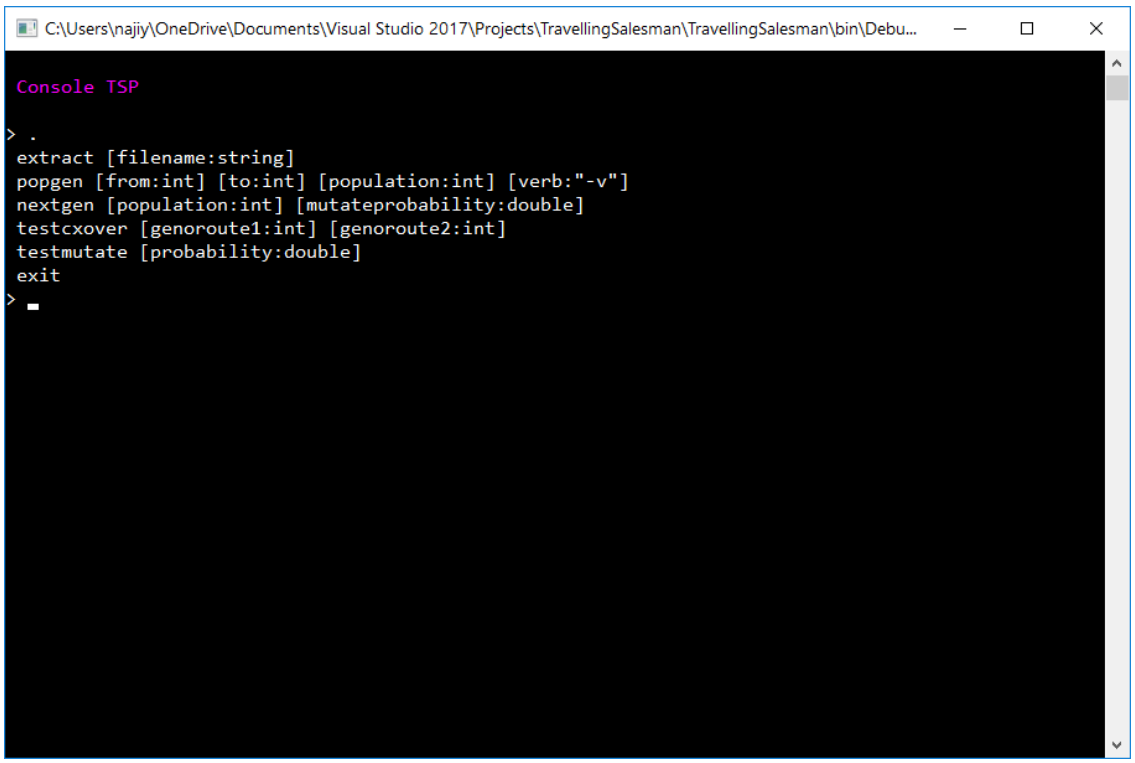


Propagating Generations

Genetic Algorithms need generation propagations. Below the command ‘nextgen 10 0.3’ yields the next generation having gone through cross-overs and mutations. The command takes the population and the mutation probability. The best result is the best route across the generations. Generation propagation below:



Commands Available



Sample Travelling Salesman Problem