**Analysis of algorithms**

**Week 9**

**Genetics Algorithms (Travel Salesman Problem)**

***Problem*:**

Given a finite number of 'cities' along with the cost of travel between each pair of them, find the shortest way of visiting all the cities and returning to your starting point

Algorithm

A traditional presentation can’t be used and algorithm for the TSP problem, because every city must be unique in a gene, and can't be duplicated.

For a gene presentation, a sequential representation is used where the cities are listed in the order in which they are visited. It's common way for TSP Genome.

Example: [9 3 4 0 1 2 5 7 6 8]

Selection

CityList1 (3 5 7 2 1 6 4 8)

CityList2 (2 5 7 6 8 1 3 4)

Fitness Function

The fitness function for the *N* cities is the sum of distances between every pair of cities in the tour.

Crossover

For a crossover operation ***Order1*** crossover will be applied.

### Mutation

We can't change the gene's bits as the usual traditional mutation does. Instead we must swap the order of cities in a path.

*Example:             Before mutation 0 1 2 3 4 5 6      After mutation     0 1 3 2 4 5 6*

Using a greedy-swap mutation. The basic idea of greedy-swap is to randomly select two cities from one chromosome and swap them if the new (swapped) tour length is shorter than the old one