National University of Computer & Emerging Sciences, Karachi Computer Science Department

Spring 2023, Lab Manual – 05

Course Code: AI-2002	Course: Artificial Intelligence Lab
Instructor(s):	Kariz Kamal, Saeeda Kanwal, Sania Urooj, Shakir Hussain, Omer Qureshi, Muhammad Ali Fatmi

Lab Tasks

Task#1:

Given a target string, the goal is to produce target string starting from a random string of the same length. In the following implementation, following analogies are made:

•Characters A-Z, a-z, 0-9 and other special symbols are considered as genes

•A string generated by these character is considered as chromosome/solution/Individual Population size= 70

•Target string to be generated: TARGET = "Artificial Intelligence Lab"

•Fitness score is the number of characters which differ from characters in target string at a particular index. So, individual having lower fitness value is given more preference.

Task#2:

Implement genetic algorithm to the Traveling Salesman Problem.

We have a set of four cities A, B, C, and D. The distances between the cities are also given to us. Here (4-1)! That is 3! Route can be generated. The tour with A B C D A will be the optimal route for given problem.

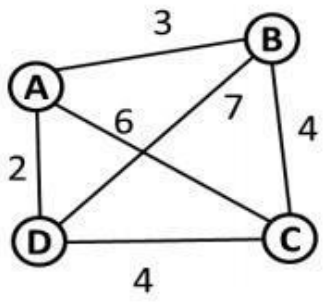

Fig. 1: Traveling salesman problem

Pseudo code

Initial Population (set of solutions) = 6

Fitness (Quality of solution) = each solution is generally represented as a string of binary numbers, known as a chromosome. The most common fitness function for TSP is the length of the route. However, the 'shorter' the route is - the better.

Crossover point, exchange data after ‘1’ instances of the list. Mutation point perform between 2nd and 4th item.

The pseudo code for genetic algorithm for implementing the Traveling salesman problem:

```
GeneticAlgorithm ( )
{
```

Initialize population of routes of cities randomly with a function Random ( ) Evaluate the fitness of each individual route using function Fitness ( )

While the fitness criteria is not satisfied do

{

Selection of two routes for reproduction using select function (Select (parent_route1, parent_route2))

Perform crossover on the selected parent routes with crossover function (child_route = Crossover (parent_route1, parent_route2))

Perform mutation on the newly generated child routes with mutation function (Mutation (child_route))

Evaluate the fitness of child_route and replace the parent population with child_route

} }

Task#3:

Write a program to solve the 8-puzzle problem using Heuristics (h(n)) for A*