CSCI 4560/6560 Evolutionary Computation

Assignment Number 2: Due 9/23/2009 **(in class)**

1. **[50 points]**

   The Subset-sum problem is stated as follows. Given a set of N positive integers $X = \{x_1, x_2, \ldots, x_n\}$ and an integer S. Find a subset P of the set X such that the sum of the elements of P is equal to S. For example, if N=5, S=21 and the set $X = \{12, 17, 3, 24, 6\}$, the set $P = \{12, 3, 6\}$ is a valid solution for the Subset-sum problem in this example.

   Use a genetic algorithm (or any evolutionary algorithm) to solve the Subset-sum problem. You should generate random integers in the range from 1 to 1000*N for the set elements. The target sum S should also be randomly chosen but you should make it at least half of the sum of all the elements or the problem will be too easy. You may download and use an existing GA or implement your own. You should try to solve the problem for the highest possible value of N that your program can handle in a reasonable amount of time (you need not go beyond N=10000). Turn in a printout of your code and your solutions. If you use an existing package, you should only turn in a printout of the parts you modify/introduce, such as the fitness function. For large values of N you need not turn in complete solution outputs.

2. **[50 points]**

   The Traveling-salesperson problem is stated as follows. Given a set of N cities, find a tour that minimizes the total distance traveled while visiting all the cities and returning to the point of origin. In this assignemnt you should use the locations of the cities in the file:

   `www.cs.uga.edu/~khaled/ECcourse/TSPDATA.txt`

   and use Euclidean distance. The file has the X and Y coordinates 127 cities.

   Use an evolutionary algorithm to solve the traveling salesperson problem. You may download and use an existing EA or implement your own. You should try to find the shortest tour among the cities. The shortest known tour has length 118282. Turn in a printout of your code and your solutions. If you use an existing package, you should only turn in a printout of the parts you modify/introduce, such as the fitness function.

Note:

- A large collection of GA and EC packages can be downloaded from the web.
- Since students will use different machines in different environments, getting a solution for a higher value of N will not be a critical factor in evaluation. However, using intelligent operators and/or strategies will earn more credit. Since this is the first assignment, a program that works correctly and solves the problem for some reasonable value of N will earn at least 80% of the grade regardless of optimality.
- You can use the same program (with some modifications) to solve both the subset-sum and the Traveling-salesperson problems. The only part that **needs** to be different is the fitness function. On the other hand, if you choose to write two different programs you are welcomed to do so as long as you finish on time.