**CS 2002 – ARTIFICIAL INTELLIGENCE PROGRAMMING**

**ASSIGNMENT NO. 2**

**Spring 2022**

Instructions

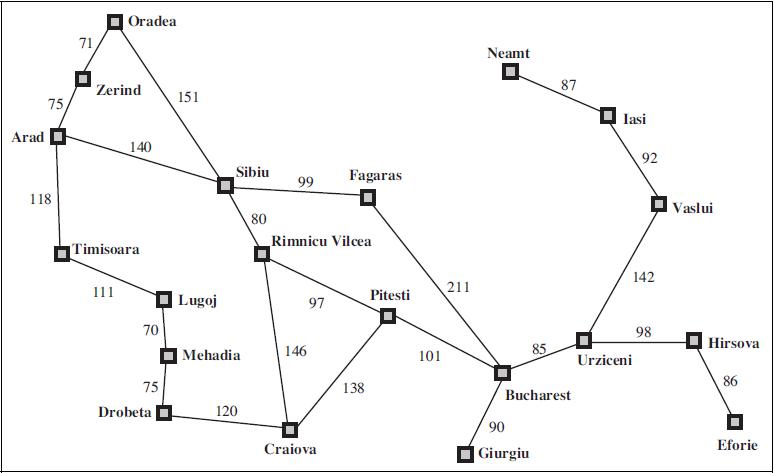
This is the first programming assignment for the course CS2002-ARTIFICIAL INTELLIGENCE in the offering spring 2022. It is suggested that you should start working on the assignment at your earliest. Question should be solved in one program file named as per suggested scheme. Your student number dash assignment number, e.g K192122-A2

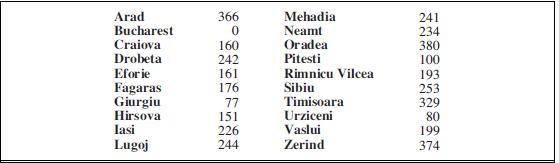
The assignment is for individual and there should not be any case of cheating. Do not share code and instruction for any problem.

Due Date: March 16th , 2022 11:00 PM

**Problem 1 Uninformed and informed Search Strategy:**

You are clear about uninformed and informed strategies now. The example discussed in the class regarding Romania map has to get implemented in this assignment.





The map is directly taken up from your book together with the heuristics table. The task is to reach from a particular source to destination using different strategies. This means that user will be facilitated with the option of choosing any random source and destination point at run time. Following are the strategies to be implemented.

1. Breadth first search
2. Uniform cost search
3. Greedy best first search
4. Iterative deepening depth first search

A comparison of these four needs to be done. Complete list of pathway and path cost of each algorithm has to be calculated so that it shows clearly that which algorithm is best out of all in ascending order.

**Problem 2: Exploring a Simple Genetic Algorithm**

In this assignment you will design and implement a simple genetic algorithm and explore its performance in evolving solution to one numerical problem, investigating the effect that various parameters have on its performance.

1. Design and implement a simple genetic algorithm, in a language of your choice, to work on binary strings, and show its performance on the Max-1s problem where the aim is to get a chromosome of all 1s. Use an 8-bit chromosome and, when calculating the fitness, map the binary number onto an integer first; so the maximum fitness for an 8-bit chromosome is 255 which is the ideal case to reach.
2. Explore the performance of the algorithm as a function of population size and number of generations. Min and max number of population size to explore performance can be of your choice for e.g. min 4 and max 10.
3. Then increase the chromosome length and investigate/explore performance as a function of this length.

By explore I mean you need to create a report in which graphs should be drawn to show how performance has varied with respect to population size, number of generations and chromosome length separately.

Bear in mind that you will be using random number generators and that, to get a representative performance and results that are statistically significant, you will need to run each experiment several times.

P.S. You are free to use any particular language for this assignment.