**MAKERERE UNIVERSITY**

**COLLEGE OF COMPUTING AND INFORMATION SCIENCE**

**DEPARTMENT OF COMPUTER SCIENCE**

**ACADEMIC YEAR 2023/2024**

**CSC 2114: ARTIFICIAL INTELLIGENCE**

**PROGRAMMING ASSIGNMENT 1**: Problem solving using search

**Due Date**: 2nd October 2023

**Instruction**

This assignment is to be done on individual basis but you are free to make consultations

**Assignment1 goal**. To enable student understand the difference between different search algorithms, their way of operation and implementation

**Problem description:** You will investigate various search algorithms for the graph in Figure 1. Edges are labeled with their costs, and heuristic values h for states are labeled next to the states. **S** is the start state, and **G** is the goal state.

**Tasks to do**

1. Read about the general operation of the following search strategies for both tree search and graph search
   1. Depth first search
   2. Breadth First Search
   3. Uniform cost search
   4. Greedy search
   5. A\* search
2. Using sets and dictionary libraries represent the graph in Figure 1 into a python graph structure.
3. Using python implement the following search strategies using for tree search
   1. Depth first search
   2. Breadth First Search
   3. Uniform cost search
   4. Greedy search
   5. A\* search
4. Using python implement the following search strategies using for graph search
   1. Depth first search
   2. Breadth First Search
   3. Uniform cost search
   4. Greedy search
   5. A\* search
5. For each of the following graph search strategies, print out the order in which states are expanded, the path returned by tree search, as well as the states that are not expanded. In all search algorithms, assume ties are broken in alphabetical order.
   1. Depth first search
   2. Breadth First Search
   3. Uniform cost search
   4. Greedy search
   5. A\* search
6. For each of the following graph search strategies, print out the order in which states are expanded, the path returned by graph search, as well as the states that are not expanded. In all search algorithms, assume ties are broken in alphabetical order
   1. Depth first search
   2. Breadth First Search
   3. Uniform cost search
   4. Greedy search
   5. A\* search

**Figure 1**