German University in Cairo
Department of Computer Science
Assoc. Prof. Haythem O. Ismail

Introduction to Artificial Intelligence, Winter Term 2019
Problem Set 2

Discussion: First Week

Exercise 2-1

Suppose we have database of video segments, covering certain topics, that may be used in the construction of presentations about said topics. In particular, the database contains three fields for every video segment:

Name, which is a unique string associated with the video segment;

Length, which is the length (in seconds) of the video segment; and

Topics, which is the non-empty set of topics covered in the video segment.

Note that a video segment may cover more than one topic and that the same topic may be covered by several video segments. Our objective is to choose a set of video segments in order to create from them a *shortest* presentation which covers a given set of topics. One way to meet this objective is to model the problem as a search problem.

a) How would you represent a state in this problem?

   Solution:

   A state is a pair $\langle T, V \rangle$, where T is a set of topics and V is a set of video segments such that the segments in V cover no topics in T.

b) What is the initial state?

   Solution:

   $\langle T_0, \{\} \rangle$, where T_0 is the given set of topics.

c) What is the goal test?

   Solution:

   $goalTest(\langle T, V \rangle)$ is true if and only if $T = \{\}$.

d) What are the operators of the problem?

   Solution:

   In state $\langle T, V \rangle$, there is one operator O_v for every video segment $v \notin V$, where $O_v(\langle T, V \rangle) = \langle T - \textbf{Topics}(v), V \cup \{v\} \rangle$.

e) What is a suitable path cost function?

   Solution:

The cost of the path to a state $\langle T, V \rangle$ may be defined as
$Cost(\langle T, V \rangle) = \sum_{v \in V} \mathbf{Length}(v)$.

Exercise 2-2

The following tree is a full search tree for some state space. Arc labels denote branch costs, double circles indicate goal nodes. For each of the following search strategies, indicate the order in which nodes will be chosen for expansion:

a) Breadth-first search.

   Solution:

   Expansion Sequence: A B C D E F G
   Path to goal: A C G

b) Depth-first search.

   Solution:

   Expansion Sequence: A B D I E J
   Path to goal: A B E J

c) Uniform-cost search.

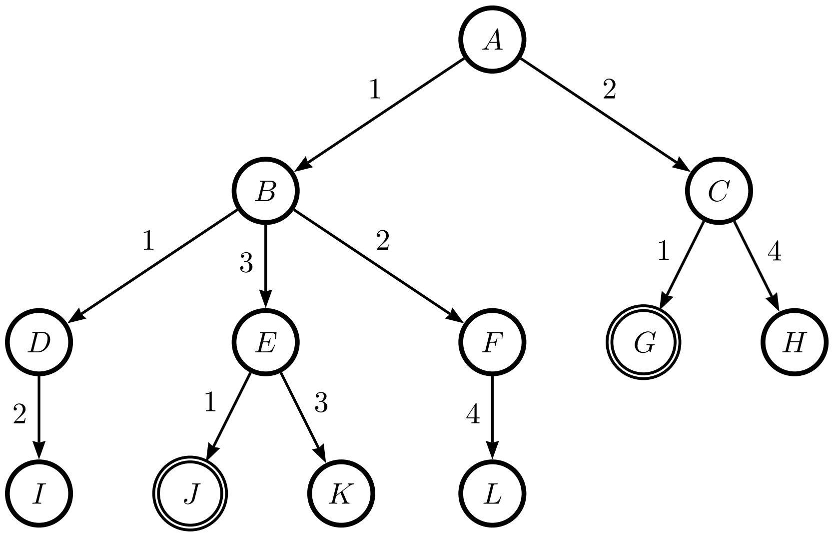
   Solution:

   Expansion Sequence: A B C D F G
   Path to goal: A C G

d) Iterative deepening search.

   Solution:

   Expansion Sequence: A A B C A B D E F C G
   Path to goal: A C G



Exercise 2-3 From R&N[1]

a) Suppose that a negative lower bound c is placed on the cost of any given step (operator) of a problem. Does this allow uniform-cost search to avoid searching the whole tree?

Solution:

If the uniform-cost algorithm finds a value which is smaller than or equal to $c \times d$, where d is the maximum depth of the tree, it can stop. Yet, such a value cannot be found except at the maximum depth. Therefore, the algorithm will not be able to save a lot of steps.

b) Suppose that there is a set of operators that form a loop, so that executing the set in some order results in no net change to the state. If all of these operators have negative cost, what does this imply about the optimal behavior for an agent in such an environment?

Solution:

For an agent who is looking for optimizing the cost function, the optimal behaviour would be to perform this set of operators in the order that produces the negative cost. The agent will execute these steps forever in order to minimize the cost.

c) One can easily imagine operators with high negative cost, even in domains such as route-finding. For example, some stretches of road might have such beautiful scenery as to far outweigh the normal costs in terms of time and fuel. Explain, in precise terms, why humans do not drive round scenic loops indefinitely, and explain how to define the state space and operators for route-finding so that artificial agents can also avoid looping.

Solution:

Humans avoid looping forever as they realize the fact that going through a scenery place is useful, yet going through it forever will not let them reach their goal. To avoid looping

[1] Russell, Stuart and Norvig, Peter (2003). Artificial Intelligence: A Modern Approach. Prentice Hall, Upper Saddle River, New Jersey.

forever in a negative cost path, add a positive cost to the transition from going through this path once to going through it again.