CPE/CSC 480 ARTIFICIAL INTELLIGENCE

MIDTERM SECTION 1

FALL 2008

PROF. FRANZ J. KURFESS

CAL POLY, COMPUTER SCIENCE DEPARTMENT

This is the Fall 2008 midterm exam for the CPE/CSC 480 class. You may use textbooks, course notes, or other material, but you must formulate the text for your answers yourself. The use of calculators or computers is allowed for viewing documents or for numerical calculations, but not for the execution of algorithms or programs to compute solutions for exam questions. The exam time is 80 minutes.

Student Name:

Signature: **Date:**

PART 1: MULTIPLE CHOICE QUESTIONS

Mark the answer you think is correct. I tried to formulate the questions and answers so that there is only one correct answer. Each question is worth 3 points.

[30 points]

a) Which statement characterizes a *rational agent* best?
- ❑ An agent that always selects the best action under the given circumstances.
- ❑ An agent that performs reasoning steps to select the next action.
- ❑ An agent that can explain its choice of an action.
- ❑ An agent that is capable of predicting the actual outcome of an action.

b) What is the key distinguishing characteristic of a learning agent (in contrast to other agent types discussed in class)?
- ❑ A learning agent can make a distinction between world states.
- ❑ A learning agent maps a real number (level-of happiness) to each state.
- ❑ A learning agent knows how to relate an action to a goal.
- ❑ A learning agent tries to improve its peformance by suggesting improvements after observing its own actions and their outcomes.

c) Which of the following agents *requires* learning capabilities?
- ❑ A conversation bot that always responds to the same question with the same answer.
- ❑ A conversation bot that, after reading the user's input, restructures its own responses to output responses using a similar vocabulary as the user, and a similar sentence structure.
- ❑ A conversation bot that responds to "yes" or "no" questions.
- ❑ A conversation bot that is able to select from possible responses in multiple-choice style questions by ranking the answers according to some evaluation function.

d) Which of the following factors is the *most important* one concerning space complexity for those search algorithms discussed in class that have exponential space complexity?
- ❑ the number of nodes already visited
- ❑ the number of nodes kept in the fringe
- ❑ the average cost of moving from one node to the next
- ❑ the memory footprint of a node (i.e. the memory required to store information about each node)

e) In comparing the Breadth First and Uniform Cost search algorithms:
- ❑ Breadth First is always better even if path costs aren't equal.
- ❑ Both are complete for non-extreme problems.
- ❑ Uniform Cost is optimal for negative cost functions.
- ❑ Uniform Cost is better for trees with an infinite branching factor.

f) Iterative Deepening A* (IDA*) is an example of a
- ❑ Algorithm whose contours always contain a finite number of nodes
- ❑ Relaxed problem
- ❑ Memory-bounded search algorithm
- ❑ Constraint satisfaction problem

g) Which of the following search algorithms is neither complete nor optimal?
- ❑ Breadth-First Search
- ❑ Depth-Limited Search
- ❑ Iterative Deepening Search
- ❑ Bi-directional Search

h) Which of the following conditions guarantees the *completeness* of the *uniform-cost search algorithm*?
- ❑ the number of branches is finite, and there are no zero-cost actions
- ❑ the path cost from the initial node to any node in the tree must be greater than zero
- ❑ the number of branches is finite, and the cost of every step (action) is greater than or equal to a small positive constant ε
- ❑ there are no branches of infinite length

i) Which of the following combinations of methods is the best characterization of *simulated annealing*?
- ❑ It is a specialization of an informed search method with specific domain knowledge from materials science (where the term "annealing" is used)
- ❑ Continually moves along the steepest gradient (either up- or downhill) and selects a random direction if it encounters a local maximum or minimum
- ❑ If it encounters a local maximum or minimum, it does a "random restart" by randomly selecting a new starting location for another search attempt
- ❑ It is a combination of gradient descent and randomized movement

j) What constitutes a *full solution* to a constraint satisfaction problem?
- ❑ the set of all variables used in the specification of the problem
- ❑ the set of all variables together with their respective potential values
- ❑ an assignment of values to *some* (but not all) variables
- ❑ an assignment of values to *all* variables

PART 2: SHORT QUESTIONS

1. Compare the expected capabilities of different agent types for a *weather forecasting agent*. The task of the agent is to predict the next day's weather, in particular the expected high and low temperatures, and the probability of rainfall. Just like commercial simple weather stations, the agent has several sensors (e.g. temperature, air pressure, and humidity), but does not have access to weather services via the Internet.

   [15 points]

   a. Use the table below to compare the properties of the agent type against the capabilities required by an agent to perform the weather forecasting task. Briefly explain your answer for each agent type. What is the least sophisticated type of agent (reflex-based, goal-based, etc.) that is reasonably capable of performing the minimal task (weather forecasting for the next day)?

Agent Type	*Description: Main Properties, Advantages, Limitations*
Reflex Agent	**Main Properties:** **Advantages:** **Limitations:**
Model-Based Agent	**Main Properties:** **Advantages:** **Limitations:**
Goal-Based Agent	**Main Properties:** **Advantages:**

	Limitations:
Utility-Based Agent	**Main Properties:** **Advantages:** **Limitations:**
Learning Agent	**Main Properties:** **Advantages:** **Limitations:**

2. **Search in the Wumpus World:** This question explores the use of different search methods for agents similar to the one in the Wumpus environment used for the lab assignments. The task of an agent is to find the goal (a tile with a special property), and the agent can move from one tile to any of four adjacent tiles that are not blocked (walls or obstacles). The configuration we assume here is a 10*10 grid, with no obstacles; you can assume that the agent starts at one corner (e.g. [0,0]), and the goal is in the opposite corner (e.g. [9,9]).

   a. What is the search time complexity for an agent using *depth-first search* in the above environment? In addition to stating the generic formula for the time complexity, please calculate a number that indicates how many search step the agent has to perform. In order to do this, you need to determine the branching factor and the depth of the tree that corresponds to this search problem. If you can not determine exact values, estimates are fine, but you should explain how you came up with the exact values or the estimates.

      10 points

   b. Are branches of infinite length a problem in this context? Please explain your answer.

      5 points

PART 3: GRAPH SEARCH

In this scenario, an agent is trying to traverse a graph from the starting point **S** to the goal point **G**. At each step, the agent can move from the current node to one of the adjacent nodes in the direction indicated by the arrow. The agent always considers alternative moves in a clockwise fashion based on the way the graph is shown below, starting from the "noon" position.

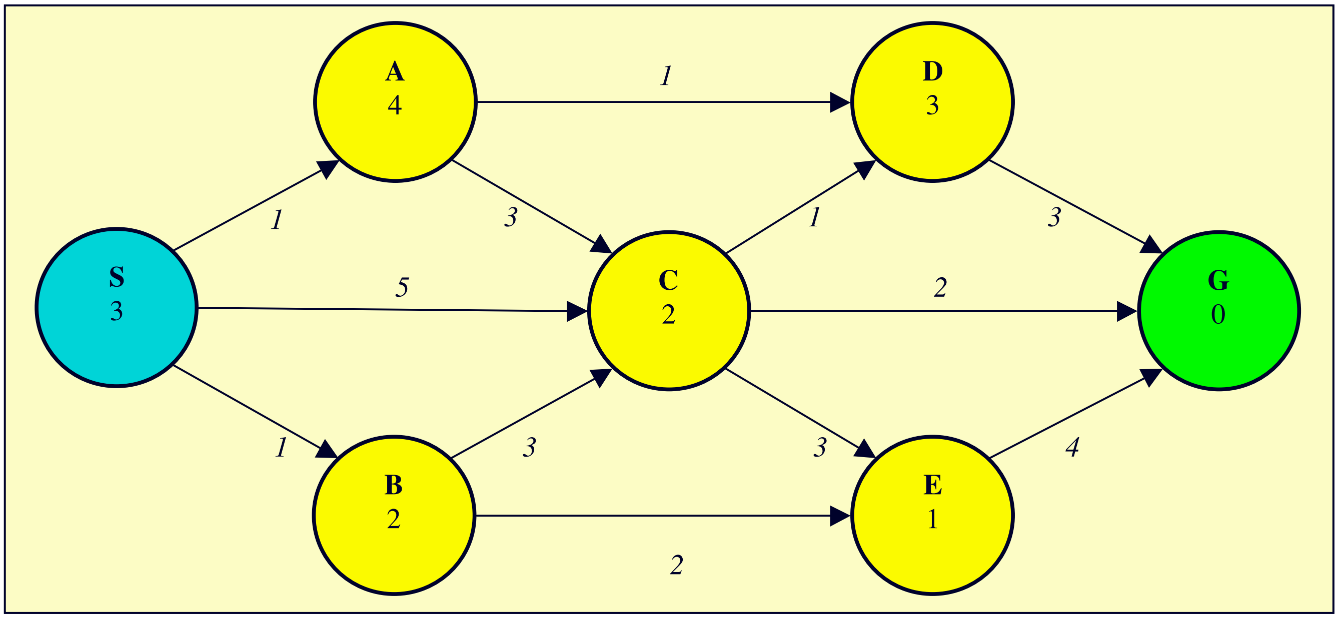
In the following parts, you need to apply different search algorithms to solve this graph traversal problem. In those algorithms that use it, calculate the path cost on the basis of the costs indicated on the edges of the graph. The heuristics to use in the respective algorithms is the number inside each node.

For the following algorithms, you need to do the following tasks:

- Mark the sequence in which the nodes are visited in the graph. You can do this directly in the copy of the graph provided for each algorithm.
- Draw the corresponding search tree. It might be advisable to draw the complete search tree first on a separate sheet of paper, and then draw the relevant parts next to / below the graph.
- Fill in the table with the information about the search trace. You may not need all the columns in the table for a given algorithm, and the size of the table does not necessarily relate to the number of steps in the algorithm. Extend the table if necessary.

a) Traverse the graph from the starting point **S** to the goal **G** according to the *Uniform-Cost Search* method.

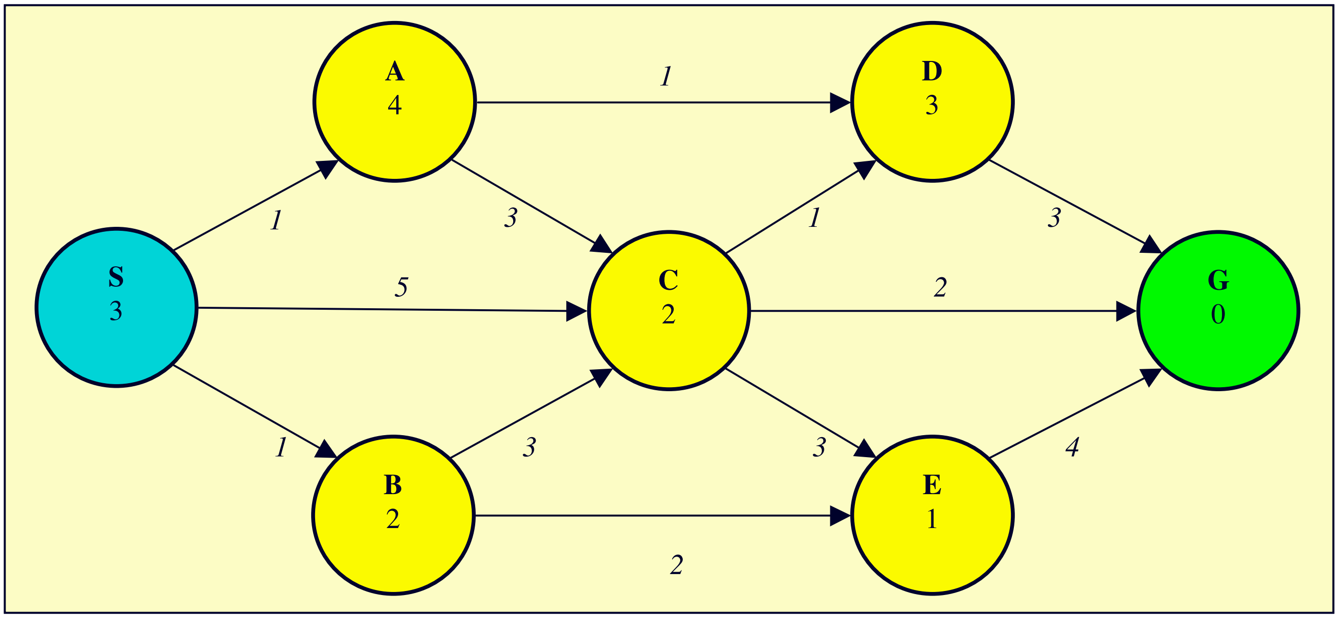
[10 points]



Step	*Current Node*	*Path Cost*	*Heuris-tic*	*F-Cost*	*Queue*
0	**S**				
1					
2					
3					
4					
5					
6					
7					
8					
9					
10					
11					
12					
13					
14					
15					
16					
17					
18					
19					
20					
21					
22					
23					
24					
25					
26					
27					
28					
29					
30					

b) Traverse the graph from the starting point **S** to the goal **G** according to the *Greedy Search* method.

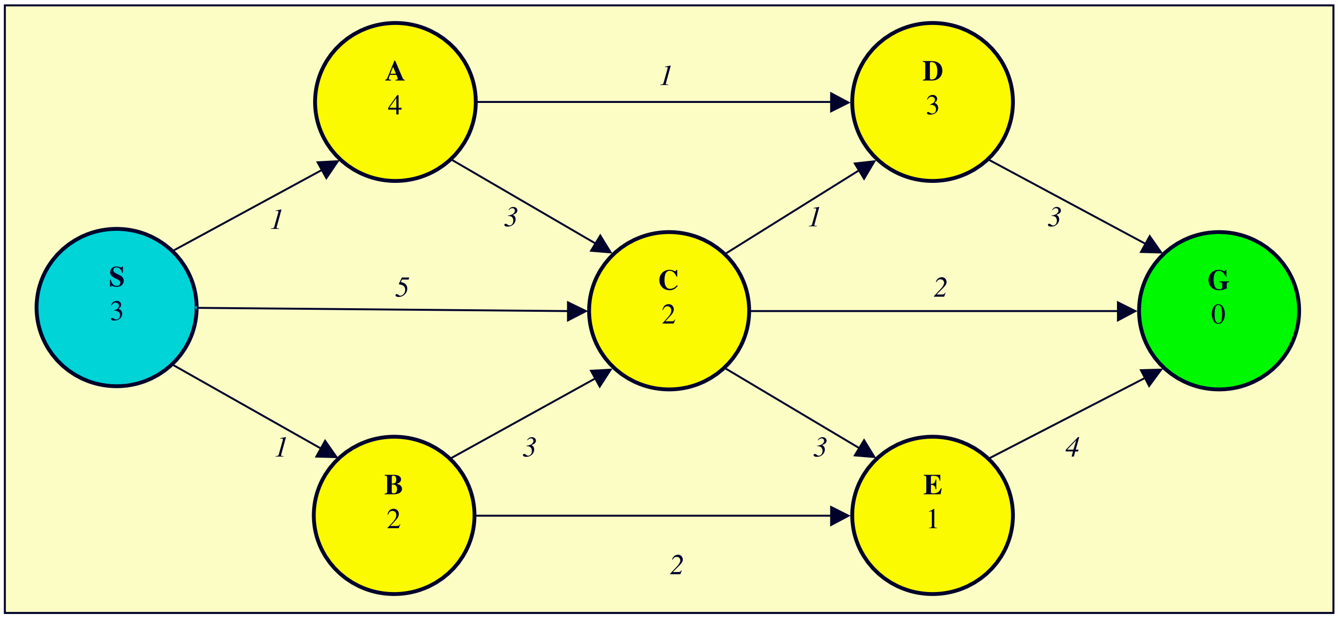
[10 points]



Step	*Current Node*	*Path Cost*	*Heuris-tic*	*F-Cost*	*Queue*
0	**S**				
1					
2					
3					
4					
5					
6					
7					
8					
9					
10					
11					
12					
13					
14					
15					
16					
17					
18					
19					
20					
21					
22					
23					
24					
25					
26					
27					
28					
29					
30					

c) Traverse the graph from the starting point **S** to the goal **G** according to the *A* Search* method. Use the simple addition f(n) = g(n) + h(n) to calculate the f-cost for the A* algorithm, with path cost and heuristic as defined earlier.

[10 points]



Step	*Current Node*	*Path Cost*	*Heuris-tic*	*F-Cost*	*Queue*
0	**S**				
1					
2					
3					
4					
5					
6					
7					
8					
9					
10					
11					
12					
13					
14					
15					
16					
17					
18					
19					
20					
21					
22					
23					
24					
25					
26					
27					
28					
29					
30					

d) Is the simple addition f(n) = g(n) + h(n) to calculate the f-cost for the A* algorithm a good choice for this problem? Explain your answer.

[5 points]

e) Is it important that the graph is directed? What would be the effect of using an undirected graph here (all other properties being the same). Explain your answer.

[5 points]

Total Points: