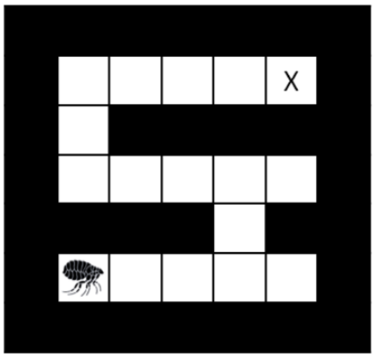
King Saud University	College of Computer and Information Sciences Department of Computer Science
CSC 361: Artificial intelligence	**Midterm exam - Fall 2019**
Date: 30/10/2019	Duration: 120 minutes

Student ID:	Name:
Sequential number:	Section/Instructor:

Question:	Problem formulation	Search	Local Search	CSP	Adversarial search	Total
Points:	20	20	20	25	15	100
Score:						

Question 1: Problem formulation .... **20 points**

Suppose you control a single flea as shown in the maze below, which must reach a designated target location **X**. At each time step, an insect can move North, East, South, or West (but not diagonally) into an adjacent square if that square is currently free, or the insect may stay in its current location. Squares may be blocked by walls (as denoted by the black squares). In addition to moving along the maze as usual, your flea can jump on top of the walls. When on a wall, the flea can walk along the top of the wall as it would when in the maze. It can also jump off of the wall, back into the maze. If the flea jumps on the wall, the path cost is increased by 2, otherwise by 1. Note that the flea can only jump onto walls that are in adjacent squares (either north, south, west, or east of the flea).



For the following questions, you should answer for a general instance of the problem, not simply for the example map shown.

1. Give a search formulation to the search problem described above.

   (a) The best state representation is:

   (A) The location of the flea as an (x, y) coordinate.

   (B) A 2D array representing the map, the position of the flea and the goal.

   (C) The location of the flea and the target X as (x, y) coordinates.

   (D) None

(b) What is the initial state:

Solution: The coordinates of the position where the flea started moving.

(c) What are the actions:

Solution: Go to the adjacent square (either north, south, east, west), stay in the same square, jump on or off a wall in adjacent squares.

(d) What is the goal state:

Solution: The bug has reached the square designated by X.

(e) The cost of an action is:

(A) The time taken by the flea to move to an adjacent cell.

(B) The number of cells visited by the flea.

(C) One for each action.

(D) None.

2. Select the heuristics that are admissible:

(A) $h1_{flea}$ = the Manhattan distance from the flea to the goal.

(B) $h2_{flea}$ = the Euclidean distance from the flea to the goal.

(C) $h3_{flea}$ = the length of the longest path from the flea to the goal.

(D) None

3. Which heuristic is better? (justify your choice):

(A) $h1_{flea}$

(B) $h2_{flea}$

(C) $h3_{flea}$

(D) None

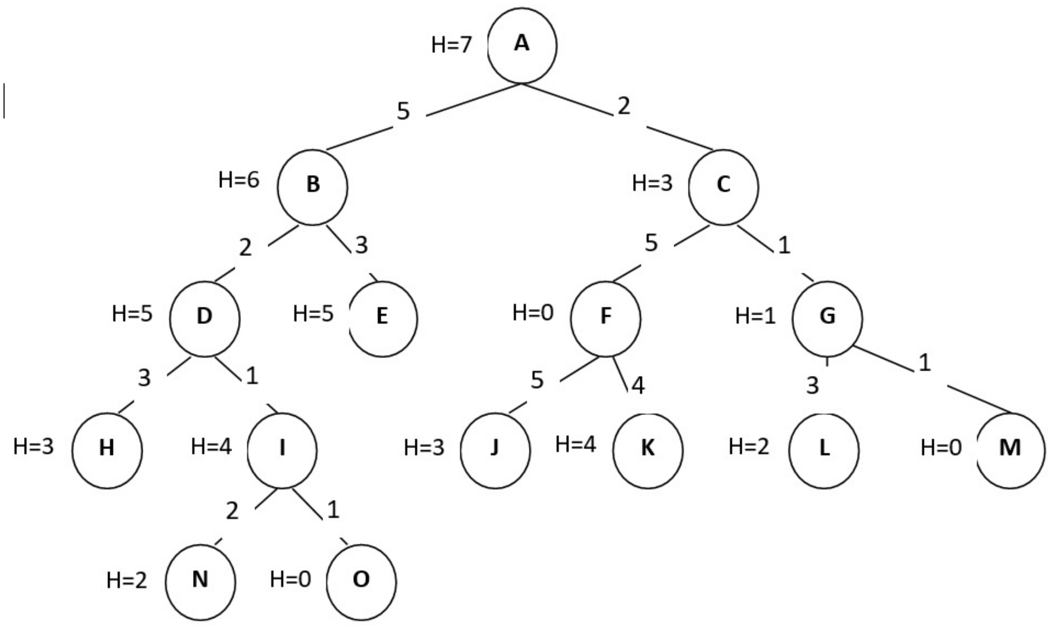
Justification:

Solution: $h1_{flea}$ dominates $h2_{flea}$

Question 2: Search .... **20 points**

(a) **Uninformed and informed search**

Consider the following graph, where A is the start state.



Given the following state space, find the goal(s) using BFS, DFS, Greedy, and A* search strategies. For each strategy, show only the order in which the nodes will be visited, the goal that was found, and the cost. Note that numbers on the arcs represent step cost and numbers to the left of each node represent the heuristic value for that node. Ties break alphabetically.

The goals have a heuristic value equal to 0.

1. After performing **BFS** graph search (**nodes are expanded in alphabetical order**):
   (a) What is the obtained goal:

   Solution: F

   (b) What is the cost of the obtained solution:

   Solution: 7

   (c) What is the order in which the nodes are visited:

   Solution: A B C D E F

2. After performing **DFS** graph search (**nodes are expanded in alphabetical order**):
   (a) What is the obtained goal:

   Solution: O

   (b) What is the cost of the obtained solution:

   Solution: 9

   (c) What is the order in which the nodes are visited:

   Solution: A B D H I N O

3. Performing **Greedy** graph search (**nodes are expanded in alphabetical order**):
   (a) What is the obtained goal:

   Solution: F

   (b) What is the cost of the obtained solution:

   Solution: 7

   (c) What is the order in which the nodes are visited:

   Solution: A C F

4. After performing **A*** graph search (**nodes are expanded in alphabetical order**):
   (a) What is the obtained goal:

Solution: M

(b) What is the cost of the obtained solution:

Solution: 4

(c) What is the order in which the nodes are visited:

Solution: A C G M

Question 3: Local Search .... 20 points

(a) Mark true or false:

1. In local search, it is essential to maintain the solution path. (A) True **(B) False**
2. In Genetic Algorithms, for reproduction, individuals are selected with a probability which is proportional to the fitness score. **(A) True** (B) False
3. The problem with the Hill Climbing algorithm is that it can get stuck in a local maximum. **(A) True** (B) False
4. A student is given a minimization problem with a search space of size 10^{10}. He decides to use Hill Climbing 1000 times, each time with a randomly selected starting point. The lowest value found is 1.3, the highest value is 4.9, and the average is 3.2. The algorithm on average takes 5 steps/loops to converge and return a result. The student considers that 1.3 as the global minimum. Is the student correct? (A) True **(B) False**
5. Genetic algorithms produce new offsprings by performing crossover or mutation. (A) True **(B) False**
6. Local search algorithms can be applied to CSP if the problem formulation is correct. **(A) True** (B) False

(b) **Problem formulation for local search**

Given a list of $n-1$ arithmetic operations in a specific order, a set S of n distinct numbers and a target number r, the goal is to find the mathematical expression, using the arithmetic operations and the numbers in S, that evaluates as close as possible (in absolute value) to r.

- No parentheses are used.
- The order of the arithmetic operations must not change.
- Each number must be used exactly once.
- The possible operations are $+, -, \times, /$.

Example 1. *Given the following expression with 2 arithmetic operations:* $|\cdots\times\cdots+\cdots|$ *and the target number* $r = 8$*, and assuming that you can use only the following distinct numbers* $S = \{1, 2, 3\}$*, the goal is to find the best numbers that can replace the dots so that the value of the expression (in absolute value) is as close as possible to* $r = 8$*.*

Some possible states are: $1 \times 2 + 3$*,* $1 \times 3 + 2$*,* $2 \times 1 + 3$*, etc. The best possible result is* $3 \times 2 + 1 = 7$ *(or* $2 \times 3 + 1$*).*

We want to solve this problem using local search (**complete state formulation**).

1. The initial state is (only one answer is correct):

   (A) an expression with no operations.

   (B) an expression with no numbers.

   (C) an empty expression.

   (D) an expression with arithmetic operations and randomly selected numbers from S.

   (E) an expression with randomly selected arithmetic operations and randomly selected numbers from S.

   (F) None.

2. Choose the appropriate set of actions for this formulation (only one answer is correct):

   (A) Starting from an expression with no numbers, add one number at each step.

   (B) Starting from an expression with no operations, add one operation at each step.

   (C) Exchange two arbitrarily chosen operations.

   (D) Exchange two arbitrarily chosen numbers.

   (E) Exchange the largest number with the smallest one.

   (F) None.

3. Let $\hat{r}$ be the result obtained in the current state. Choose the appropriate objective function (only one answer is correct):

   (A) $\hat{r}$

   (B) $|n - \hat{r}|$

   (C) $r - \hat{r}$

   (D) $|r - \hat{r}|$

   (E) $\hat{r}/4$

   (F) None

Question 4: CSP .... **25 points**

(a) **Problem formulation for CSP**

Consider the problem of Sudoku where:

- No value appears twice in the same row.
- No value appears twice in the same column.
- No value appears twice in the same block (2X2 block)

The figure shows the Sudoku. Cells should be filled with values within $1, 2, 3, 4$

V_1	3	V_2	1
V_3	1	V_4	4
3	4	1	2
V_5	V_6	4	V_7

1. Give a formulation to the CSP problem (**use only binary constraints**):

Solution: Variables: V1, V2, V3, V4, V5, V6, V7

Domains: 1,2,3,4 OR D1=2,4, D2=2, D3=2, D4=2,3, D5=1,2, D6=2, D7=3. Note: if first answer is chosen then the illegal values must restricted by the constraints.

Constraints: 1. $V1 <> V3$

2. $V1 <> V5$

3.$V1 <> V2$

4. $V2 <> V4$

5. $V3 <> V4$

6. $V3 <> V5$

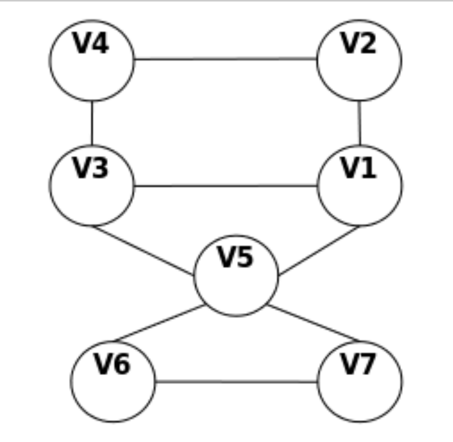
7. $V5 <> V6$

8. $V5 <> V7$

9. $V6 <> V7$

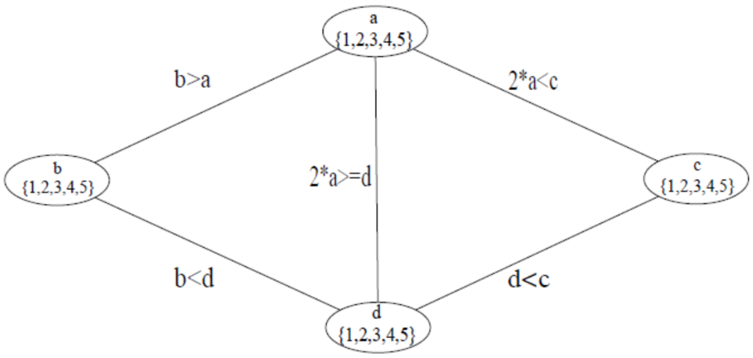
2. Draw the corresponding constraint graph. **The edges must not intersect**.

Solution:



(b) **Applying AC3**

Apply AC3 algorithm to the following graph.



Queue	a $\{1,2,3,4,5\}$	b $\{1,2,3,4,5\}$	c $\{1,2,3,4,5\}$	d $\{1,2,3,4,5\}$	Added arcs
ab					
ba					
ac					
ca					
ad					
da					
bd					
db					
cd					
dc					

Solution:

Arcs	a	b	c	d	Added arcs
ab	{1, 2, 3, 4}				ca, da
ba		{2, 3, 4, 5}			db
ac	{1, 2}				ba, da
ca			{3, 4, 5}		dc
ad	-				-
da				{1, 2, 3, 4}	bd, cd
bd		{2, 3}			ab
db				{3, 4}	ad, cd
cd			{4, 5}		ac
dc				-	-
ba		-			-
ab	-				-
ad	{2}				ba, ca
ac	-				-
ba		{3}			db
ca			{5}		dc
db				{4}	ad, cd
dc				-	-
ad	-				-
cd			-		-

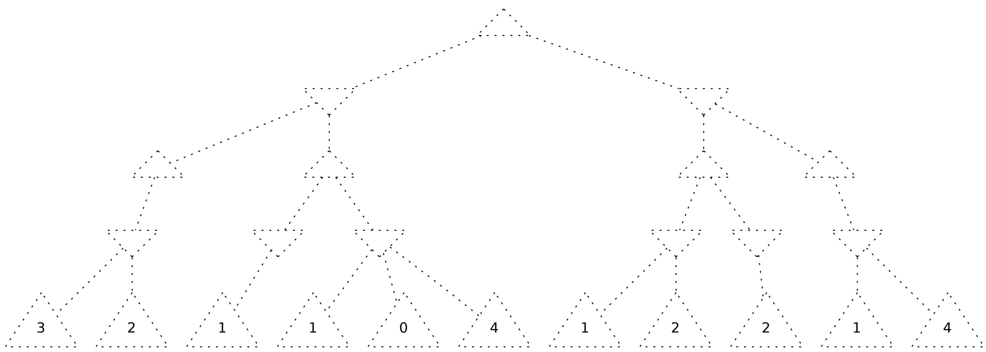
After performing AC3, propose a solution to the CSP problem.

Solution:

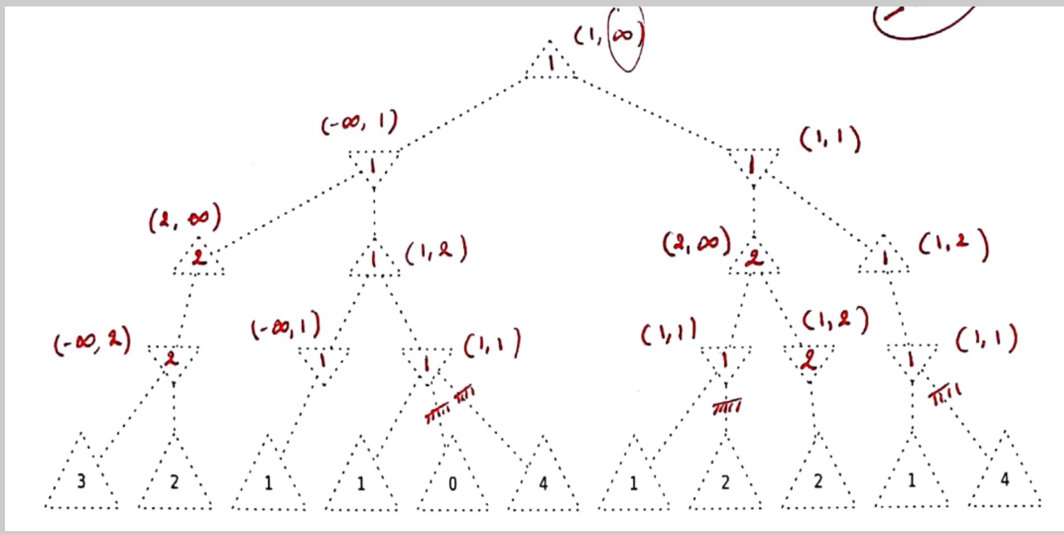
a=2, b=3, c=5, d=4

Question 5: Adversarial search .... 15 points

Apply $\alpha-\beta$ to the following game tree (indicate the **final** values of α and β and the value of each nonterminal node):



Solution:


(1, ∞)
1
(-∞, 1)
1
(1, 1)
1
(2, ∞)
2
(1, 2)
1
(2, ∞)
2
1
(1, 2)
(-∞, 2)
2
(-∞, 1)
1
(1, 1)
1
(1, 1)
1
(1, 2)
2
1
(1, 1)
3
2
1
1
0
4
1
2
2
1
4