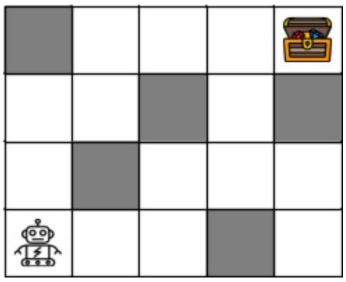
Intelligent Systems – Exam Block 1, November 3, 2023
Test A (1.75 points) <u>score</u> : max (0, (corrects – errors/3)*1.75/9)

Surnames : **Name** :
Group : A B C D E F G 4IA

1) The figure shows a board where the robot wants to reach the treasure and the shaded squares are obstacles. The robot can move to a square up, down, right or left (all movements have cost 1) as long as it does not go outside the boundaries of the board and does not move to a square containing an obstacle. The robot applies an A algorithm with the heuristic h(n) = Manhattan distance, the same heuristic as for the 8-puzzle. Indicate the **<u>CORRECT</u>** answer :



A. For every state n of the problem, it holds h(n)<h*(n).
B. The A algorithm will not find the optimal solution.
C. If in addition to the 4 indicated movements, the robot can jump over an obstacle (cost=1), h(n) would not be admissible.
D. An admissible heuristic cannot be defined for this problem.

---

2) Given the initial *Facts* ={(list 6 5 9 0 4 4 3) (minimum 10)} and the following rule to calculate the minimum of a list:

```
(defrule r1
?f1 <- (list $?a ?b $?c)
?f2 <- (minimum ?m)
        (test (< ?b ?m))
=>
(assert (minimum ?b))
(retract ?f2))
```

If our goal is to obtain the final *Facts* (after successive execution of the rule) in which the 'minimum' fact can only appear once (containing the minimum value of the list). Which of the following statements is **<u>TRUE</u>** to achieve our goal?

A. It would be necessary to add (retract ?f1)
B. It would be necessary to remove (retract ?f2)
C. It would be necessary to change the test by adding (test (> ?b ?m))
D. The rule is correct

3) A fact in CLIPS like (puzzle 2 8 3 1 6 4 7 0 5) is a linear representation of a 3x3 board of the 8-puzzle configuration that can be seen in the figure below:

2	8	3
1	6	4
7		5

Let the following rule be in CLIPS:

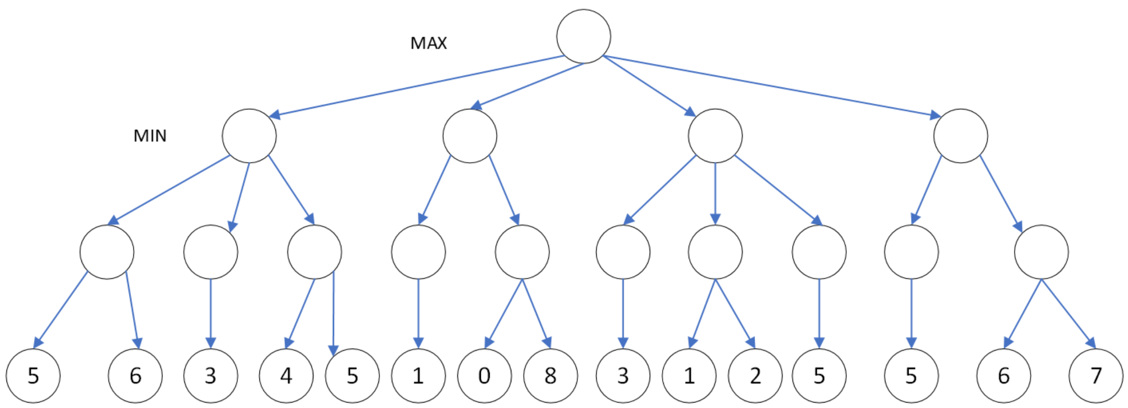
```
(defrule R1
?f<-(puzzle $?x ?a ?b ?c 0 $?y)
=>
(retract ?f)
(assert (puzzle $?x 0 ?b ?c ?a $?y)))
```

Indicate the **CORRECT** answer:

A. The rule performs a correct move of a non-blank tile down.
B. The rule performs a correct move of a non-blank tile up.
C. For the rule to be a correct move of a non-blank tile down, a test would have to be added to check that the blank tile is not in the top row.
D. For the rule to be a correct move of a non-blank tile up, a test would have to be added to check that the blank tile is not in the bottom row.

---

4) How many nodes do not need to be generated using the alpha-beta algorithm compared to the Minimax algorithm in the following tree (left-to-right expansion)?



A. 9
B. 10
C. 8
D. 11

---

5) If the IDA* algorithm is applied using a heuristic function h(n), the number of iterations will depend on:

A. Whether the function h(n) is consistent or not.
B. The number of different values of f(n) that are generated during the search.
C. Whether backtracking is used in the search or not.
D. None of the above answers is correct.

---

6) Given the fact:

(servers server 1 cpu 5 gpu 3 server 2 cpu 7 gpu 5 server 3 cpu 10 gpu 4)

where the number after server is the identifier of the server and the numeric values after cpu and gpu indicate the number of CPUs and GPUs the server has. Which of the following patterns would allow us to obtain the identifier of a server and the number of GPUs that said server has ?

A. (servers server ?s $? gpu ?g $?)
B. (servers $? server ? cpu $? gpu ?g $?)
C. (servers server ?s cpu ? gpu ?g)
D. (servers $? server ?s cpu ? gpu ?g $?)

---

7) According to the following RBS, what would be the final state of *Facts* after its execution?

```
(deffacts testing
    (A) (B) (C) (D))

(defrule R1
    ?f1 <- (A)
           (B)
 =>
    (assert (E))
    (retract ?f1))

 (defrule R2
    ?f1 <- (C)
           (E)
 =>
    (assert (A))
    (retract ?f1))
```

A. *Facts* = {(A) (B) (C) (D)}
B. *Facts* = {(A) (B) (C) (D) (E)}
C. *Facts* = {(B) (D) (E)}
D. The execution never ends, since R1 and R2 are executed continuously

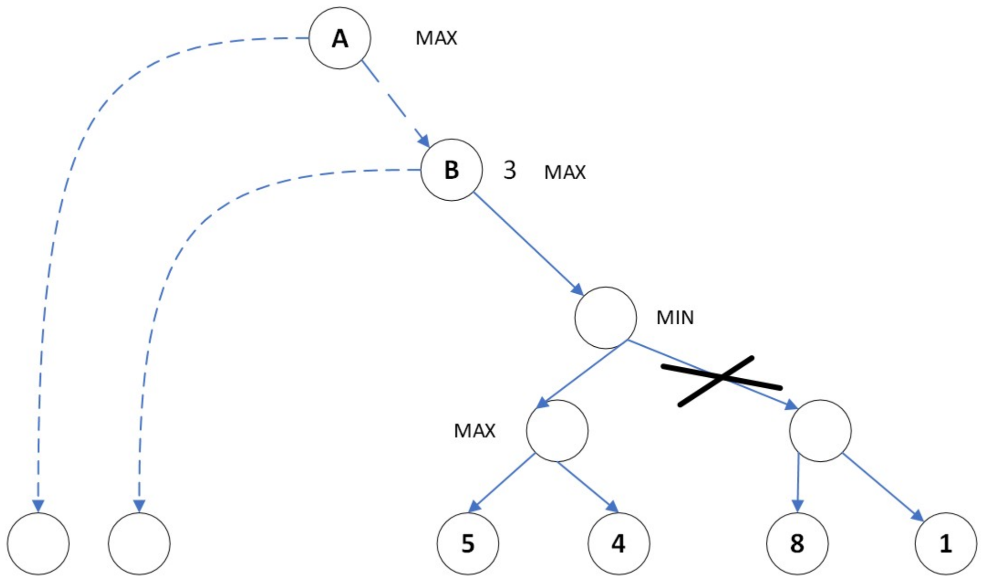
---

8) Which statement regarding the RBFS algorithm is **INCORRECT**?:

A. RBFS expands first nodes with lower f-value.
B. All nodes generated in the search tree store the f-value of its second best child.
C. RBFS updates the stored value of a node with the minimum between the stored value of its best sibling and the bound provided by its parent.
D. Unlike the IDA* algorithm, RBFS does not always have to regenerate the entire tree from the root node when the bound is exceeded.

---

9) Considering that node B has a provisional minimax value of 3, what provisional minimax value should node A have for the indicated cut to occur?

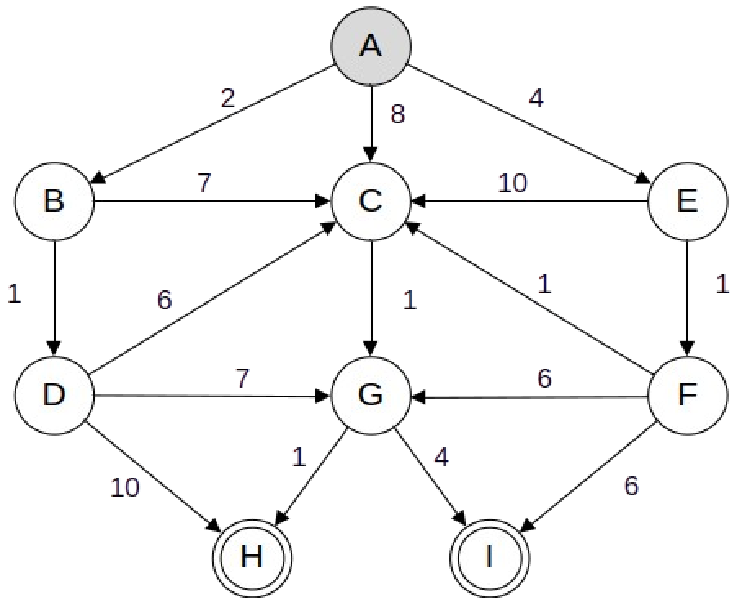


A. Greater than or equal to 3
B. Greater than or equal to 4
C. Greater than or equal to 5
D. Greater than or equal to 8

Intelligent Systems – Exam Block 1, November 3, 2023
Problem: 2 points

The following graph represents a state space problem. The nodes of the graph are the states of the problem, the edges connect each state to its successors, and the numerical value of each edge represents the cost of moving from one state to the corresponding successor. The initial state of the problem is node A and the goal states are H and I. The heuristic function h(n) is indicated in the table:

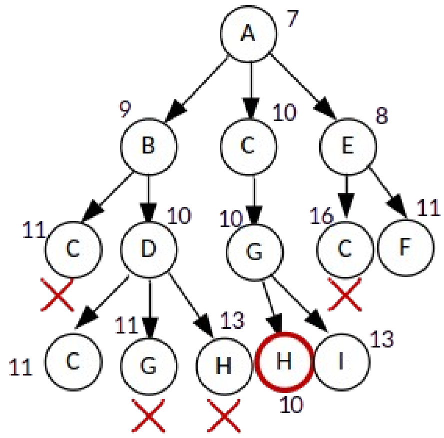
n	**A**	**B**	**C**	**D**	**E**	**F**	**G**
h(n)	7	7	2	7	4	6	1



1) (0.7 points) Draw the tree that is generated when performing a search with an A algorithm with control of repeated nodes only in OPEN (TREE-SEARCH search) indicating the states in OPEN when it changes, the solution path obtained and its cost. Given a draw in f-value, choose the nodes in alphabetical order.
2) (0.3 points) Is the answer found in the previous question the optimal solution? Is the heuristic admissible? and consistent? Justify your answers.
3) (0.6 points) The application of an ID algorithm (based on DFS search with TREE-SEARCH), applying control of repeated nodes in OPEN, what solution would it find and how many iterations does it need to find the solution? What is the maximum number of nodes stored in memory? Justify your answers.
4) (0.4 points) Suppose that the vertices in the graph above represent different locations and the edge costs represent the distance in meters that must be traveled from one location to another. We know that a robot is at a location X and it has traveled a distance less than or equal to 5 meters to reach node X. We also know that the estimated cost from X to reach a goal location is one meter lower than the actual cost to reach this goal location. Which node is X in the graph? Which goal location will it reach? Justify your answers.

SOLUTION:

1)



OPEN={A}
OPEN={E(8), B(9), C(10)}
OPEN={B(9), C(10), F(11)} ;; C(16) node repeated in OPEN
OPEN={C(10), D(10), F(11)} ;; C(11) node repeated in OPEN
OPEN={D(10), G(10), F(11)}
OPEN={G(10), C(11), F(11), H(13)};;node G(11) repeated in OPEN, we insert node C
OPEN={H(10), C(11), F(11), I(13)} ;;
OPEN={C(11), F(11), I(13)}

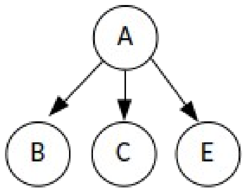
Solution: A-C-G-H Cost: 10

2) No, it is not the optimal solution, since there is a solution with cost 8: A-E-F-C-G-H . The heuristic function is non-admissible because h(F) = 6 > h*(F) = 3, and it is also not consistent because h(F) <= h(C) + 1 (6 <= 2+1) is not satisfied.

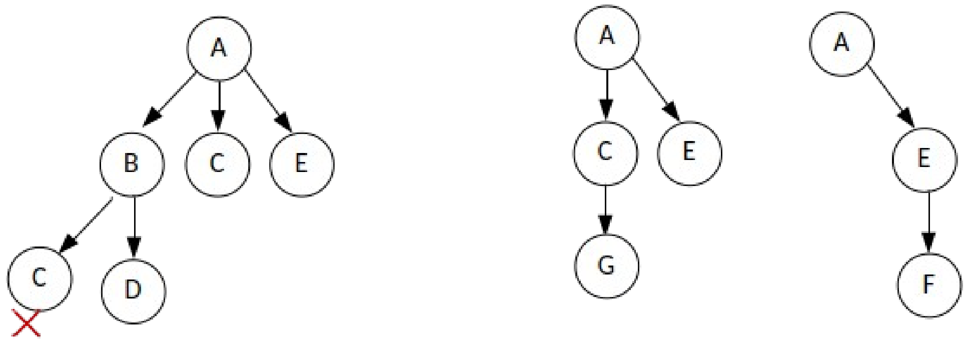
3) The ID needs 4 iterations and would also find solution H. The maximum number of nodes stored in memory is reached in the last iteration, being 7, if the path to the goal node has to be returned, or 4, if just the goal node is returned:

Iter 1

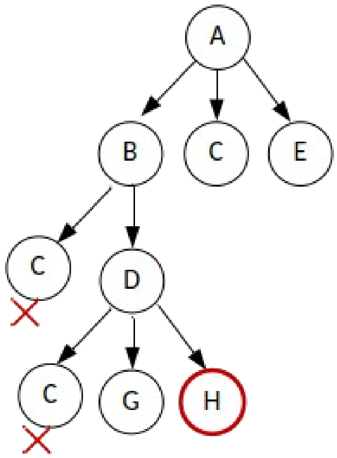
Iter 2



Iter 3



Iter 4



4) We know that g(X) <= 5, so X is included in {B,D,E,F}. Also, h(X) = h*(X) - 1. If we compare the actual cost with the estimated cost for these nodes:

h(B)= 7, h*(B)= 9
h(E)= 4, h*(E)= 7
h(D)= 7, h*(D)= 8
h(F)= 6, h*(F)= 6

Then, the robot is located in D, since h(D) = h*(D) – 1. The goal location that will be reached applying an A search is H.

Assuming that the robot has not traveled any distance yet, then it would be at node A, where h(A)=7 and h*(A)=8 satisfying the restrictions. In this case, it would reach the goal node H applying an A search, as shown in the first exercise.