CS-171, Intro to A.I. — Quiz#2 — Fall Quarter, 2012 — 20 minutes
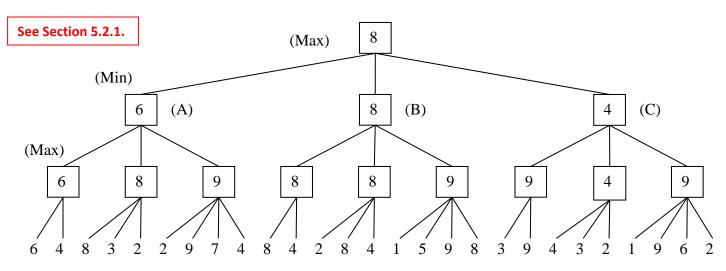
YOUR NAME AND EMAIL ADDRESS: _____

YOUR ID: _____ ID TO RIGHT:_____ ROW:_____ NO. FROM RIGHT:_____

1. (25 pts total, -5 pts for each error, but not negative) MINI-MAX SEARCH IN GAME TREES.
The game tree below illustrates a position reached in the game. Process the tree left-to-right. It is **Max**'s turn to move. At each leaf node is the estimated score returned by the heuristic static evaluator.
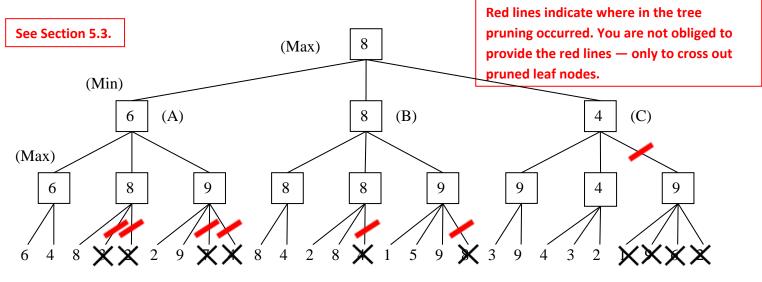
1.a. Fill in each blank square with the proper mini-max search value.

1.b. What is the best move for Max? (write A, B, or C) _____ B _____

See Section 5.2.1.

(Max) 8

(Min)

6 (A) 8 (B) 4 (C)

(Max)

6 8 9 8 8 9 9 4 9

6 4 8 3 2 2 9 7 4 8 4 2 8 4 1 5 9 8 3 9 4 3 2 1 9 6 2

2. (25 pts total, -5 for each error, but not negative) ALPHA-BETA PRUNING. Process the tree left-to-right. This is the same tree as above (1.a). You do not need to indicate the branch node values again.

Cross out each leaf node that will be pruned by Alpha-Beta Pruning.

See Section 5.3.

Red lines indicate where in the tree pruning occurred. You are not obliged to provide the red lines — only to cross out pruned leaf nodes.

(Max) 8

(Min)

6 (A) 8 (B) 4 (C)

(Max)

6 8 9 8 8 9 9 4 9

6 4 8 X X 2 9 X X 8 4 2 8 X 1 5 9 X 3 9 4 3 2 X X X X

**** TURN PAGE OVER AND CONTINUE ON THE OTHER SIDE ****

3. (50 pts total, 10 pts eac[h ... ...]ember visited nodes, so
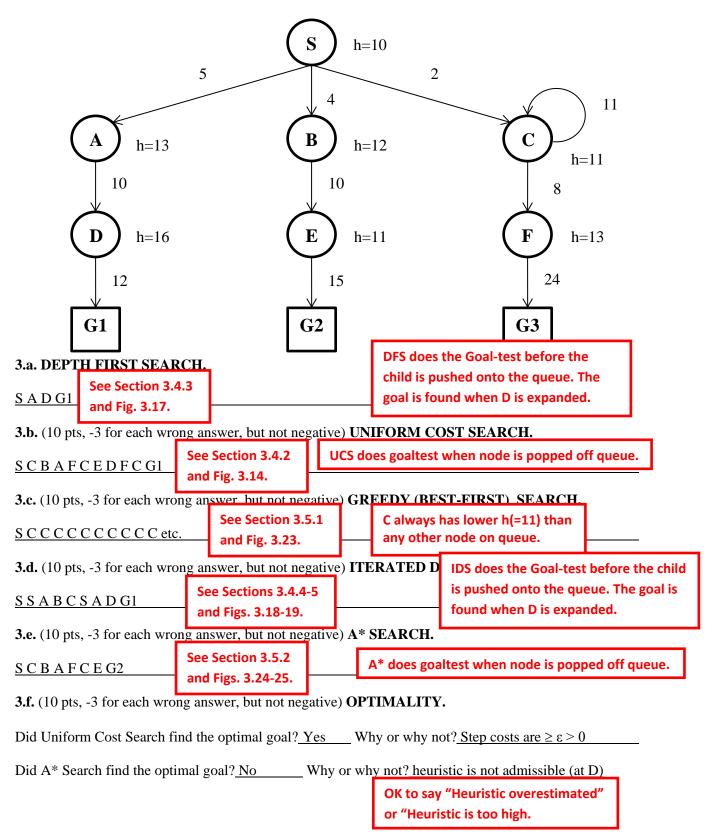repeated nodes are possibl[e ...].

Step costs are given [... ...]de (as h=x). The
successors of each node are indicated by the arrows out of that node. Successors are returned in left-to-right
order. **(Note: C is a successor of itself).**

For each search strategy below, indicate the order in which nodes are expanded (i.e., to expand a node
means that its children are generated), ending with the goal node that is found.

The first one is done for you as an example.

```
                          ( S )   h=10

          5                      4                      2          11
                                                                  ___
      ( A )  h=13            ( B )  h=12            ( C )─────────┘
                                                      h=11
          10                     10                     8

      ( D )  h=16            ( E )  h=11            ( F )  h=13

          12                     15                     24

      [ G1 ]                 [ G2 ]                 [ G3 ]
```

3.a. DEPTH FIRST SEARCH.

S A D G1

**DFS does the Goal-test before the
child is pushed onto the queue. The
goal is found when D is expanded.**

3.b. (10 pts, -3 for each wrong answer, but not negative) **UNIFORM COST SEARCH.**

S C B A F C E D F C G1

UCS does goaltest when node is popped off queue.

3.c. (10 pts, -3 for each wrong answer, but not negative) **GREEDY (BEST-FIRST) SEARCH.**

S C C C C C C C C C C etc.

**C always has lower h(=11) than
any other node on queue.**

3.d. (10 pts, -3 for each wrong answer, but not negative) **ITERATED D[...]**

S S A B C S A D G1

**IDS does the Goal-test before the child
is pushed onto the queue. The goal is
found when D is expanded.**

3.e. (10 pts, -3 for each wrong answer, but not negative) **A* SEARCH.**

S C B A F C E G2

A* does goaltest when node is popped off queue.

3.f. (10 pts, -3 for each wrong answer, but not negative) **OPTIMALITY.**

Did Uniform Cost Search find the optimal goal? Yes   Why or why not? Step costs are $\geq \varepsilon > 0$

Did A* Search find the optimal goal? No   Why or why not? heuristic is not admissible (at D)

**OK to say "Heuristic overestimated"
or "Heuristic is too high."**

3.e. A* SEARCH. Represent nodes as state/h/g/f.

Initial Queue: (S/10/0/10)   .

Popped Node: (S/10/0/10)   .

Children (left-to-right): (A/13/5/18) (B/12/4/16) (C/11/2/13)   .

Queue (sorted): (C/11/2/13) (B/12/4/16) (A/13/5/18)   .

Popped Node: (C/11/2/13)   .

Children: (C/11/13/24) (F/13/10/23)   .

Queue: (B/12/4/16) (A/13/5/18) (F/13/10/23) (C/11/13/24)   .

Popped Node: (B/12/4/16)   .

Children: (E/11/14/25)   .

Queue : (A/13/5/18) (F/13/10/23) (C/11/13/24) (E/11/14/25)   .

Popped Node: (A/13/5/18)   .

Children: (D/16/15/31)   .

Queue: (F/13/10/23) (C/11/13/24) (E/11/14/25) (D/16/15/31)   .

Popped Node: (F/13/10/23)   .

Children: (G3/0/34/34)   .

Queue : (C/11/13/24) (E/11/14/25) (D/16/15/31) (G3/0/34/34)   .

Popped Node: (C/11/13/24)   .

Children: (C/11/24/35) (F/13/21/34)   .

Queue : (E/11/14/25) (D/16/15/31) (G3/0/34/34) (F/13/21/34) (C/11/24/35)   .

Popped Node: (E/11/14/25)   .

Children: (G2/0/29/29)   .

Queue : (G2/0/29/29) (D/16/15/31) (G3/0/34/34) (F/13/21/34) (C/11/24/35)   .

Popped Node: (G2/0/29/29)   .

Children: Goal test succeeds.   .