first name _____ **last name** _____ **std.id#** _____

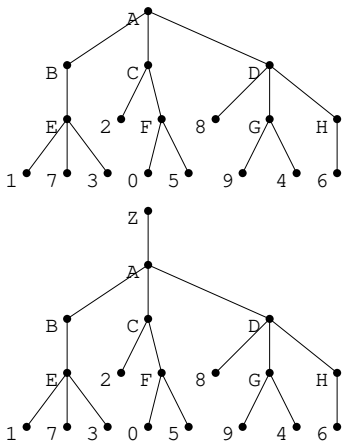
12 marks **30 min (× 1.5)** **closed book** **no devices** **1 page**

1. [3 marks] For tree at right, the root is a MAX node, leaf scores are for MAX. Give minimax values below.

   A 7 B 7 C 2 D 6
   E 7 F 5 G 9 H 6

   For tree at right, the root is a MAX node, leaf scores are for MAX. Give minimax values below.

   Z 1
   A 1 B 1 C 2 D 8



   For tree at left, the root is a MIN node, leaf scores are for MAX. Give minimax values below.

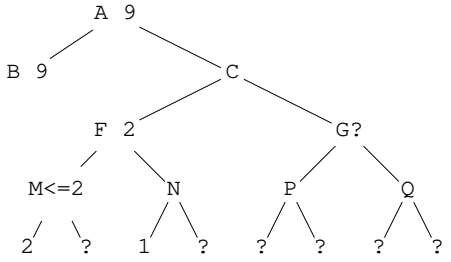
   A 1 B 1 C 2 D 8
   E 1 F 0 G 4 H 6

2. [3 marks] Fill in blanks (code from `negamax.py`). (d depth, G game tree, V terminal node scores, w node)

```
def negamax(d, G, V, w):
  if isTerminalNode(w,V): return V[w]
  val = NEGINF
  for c  in G[w]: # for each child c  of v

    val = max(val, -negamax(d+1, G, V, c))
  return val
```

3. [3 marks] In this alphabeta search, subtrees of M,N,C were pruned. Explain each prune: fill in the blanks.



   after search completes, bound at node M is shown,

   bound at node N is <= 1, bound at node C is <= 2

   M: on path-to-root M-F-C-A, MAX has move option A-B

   so alpha =9, MIN has move option M-2 so beta =2

   C: on path-to-root C-A, MAX has move option A-B

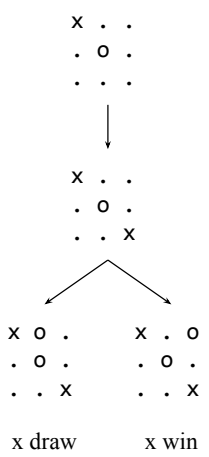
   so alpha =9, MIN has move option C-F so beta =2

4. [3 marks] For this position with x to play, draw the next two levels of a proof tree that shows that x can at least draw: at level 1 of the tree you can prune isomorphic positions. Below each leaf of your drawing, give the x-minimax value (win/lose/draw) of that subtree.

ROUGH WORK HERE

```
x . .
. o .
. . .
```

FINAL ANSWER



first name _____ **last name** _____ **std.id#** _____

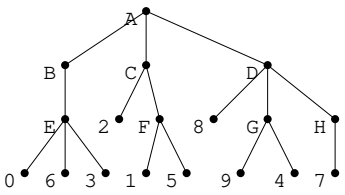
12 marks **30 min (× 1.5)** **closed book** **no devices** **1 page**

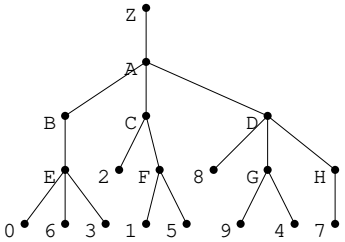
1. [3 marks] For tree at right, the root is a MAX node, leaf scores are for MAX. Give minimax values below.

   A 7 B 6 C 2 D 7
   E 6 F 5 G 9 H 7

   For tree at right, the root is a MAX node, leaf scores are for MAX. Give minimax values below.

   Z 0
   A 0 B 0 C 2 D 8





For tree at left, the root is a MIN node, leaf scores are for MAX. Give minimax values below.

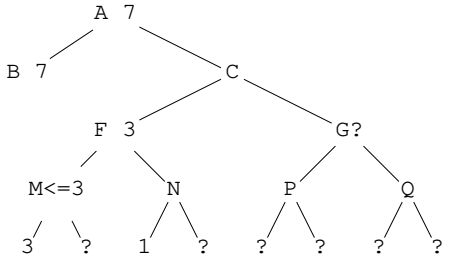
A 0 B 0 C 2 D 8
E 0 F 1 G 4 H 7

2. [3 marks] Fill in blanks (code from `negamax.py`). (q depth, T game tree, S terminal node scores, v node)

```
def negamax(q, T, S, v):
  if isTerminalNode(v,S): return S[v]
  val = NEGINF
  for ch in T[v]: # for each child ch of v

    val = max(val , -negamax(q+1, T, S, ch))
  return val
```

3. [3 marks] In this alphabeta search, subtrees of M,N,C were pruned. Explain each prune: fill in the blanks.



after search completes, bound at node M is shown,

bound at node N is <=1, bound at node C is <=3

M: on path-to-root M-F-C-A, MAX has move option A-B

so alpha =7, MIN has move option M-3 so beta =3

C: on path-to-root C-A, MAX has move option A-B

so alpha =7, MIN has move option C-F so beta =3

4. [3 marks] For this position with `x` to play, draw the next two levels of a proof tree that shows that `x` can at least draw: at level 1 of the tree you can prune isomorphic positions. Below each leaf of your drawing, give the x-minimax value (win/lose/draw) of that subtree.

ROUGH WORK HERE

```
. . .
. o .
x . .
```

FINAL ANSWER