1. [1 marks] **Complete this page properly.**

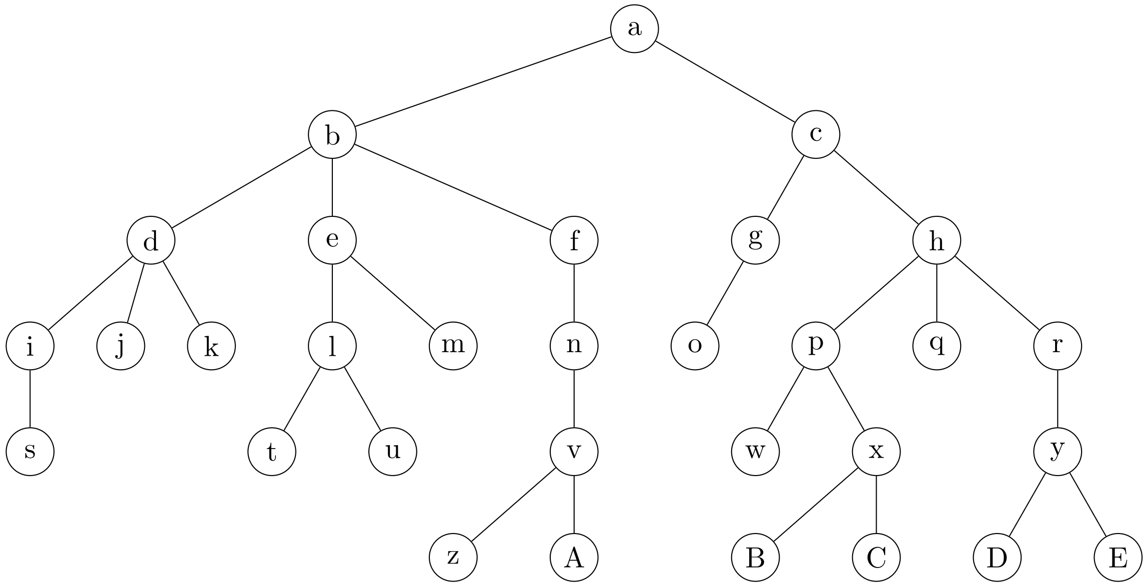
   Each group submits only one assignment. Print the name and ID number of each group member (at most 4) for this assignment:

   Acknowledge **all** sources, including all references and all people not in your group with whom you discussed any part of any question (for each discussion, list the relevant questions) (continue on the back of this page if there is insufficient space). If none, write *webnotes and lectures only*.

   Each group member must read, agree to, and sign this statement:

   I am familiar with the Code of Student Behaviour. I understand that there are significant penalties for any infraction of this Code, including failure to acknowledge sources. I have not shared any written or printed version of any of my answers with any other student.

2. [4 marks]



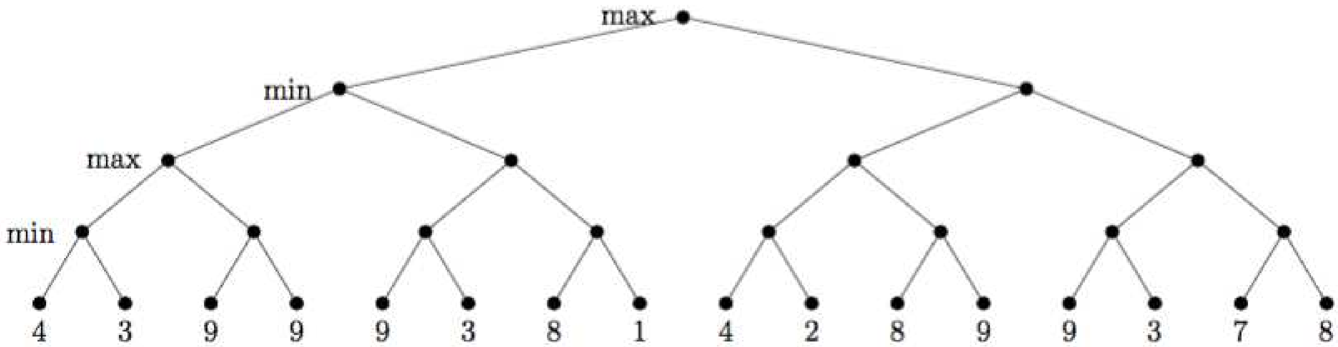
In this game tree, for first-player max, find minimax values for each node. Leaf nodes $s, j, k, t, u, m \ldots D, E$) have max-scores 8, 7, 3, 9, 2, 5, 7, 6, 4, 1, 0, 3, 2, 5, 7 respectively.

node	a	b	c	d	e	f	g	h	i	j	k	l	m	n	o	p	q	r	s	t	u	v	w	x	y	z	A	B	C	D	E
value																															

Using negamax, for the same game tree, label each node with the score of the player-to-move: recall that the score for min is the negative of the score for max. Scores at even (resp. odd) depth are for max (resp. min). E.g. leaf s has depth 4, so is a max node, so label s with max-score 8; its parent node i is a min node, so label i with min-score -8. E.g. D, E are min leafs, so label them $-5, -7$, their parent y is a max node, so label y $\texttt{max}\{--5, --7\} = 7$.

node	a	b	c	d	e	f	g	h	i	j	k	l	m	n	o	p	q	r	s	t	u	v	w	x	y	z	A	B	C	D	E
value																															

3. [2 marks] As in last exercise in `http://webdocs.cs.ualberta.ca/~hayward/396/asn/mmx.pdf`, label each search tree node with its final (not negamax) α-β value: either the minimax value (if found), or the latest lower or upper bound (if found), or nothing (if the node was never visited). Hint: check your answer using `alphabeta.py` in class repo `games-puzzles-algorithms/simple/alphabeta`.



4. [2 marks] We can represent a tic-tac-toe game by its move sequence. Label cells 1 ... 9, row by row, starting from the top, as shown below. The move sequence 1.X[4] 2.O[5] 3.X[8] 4.O[7] 5.X[2] 6.O[3] — or simply (4 5 8 7 2 3) — leaves the position below. So, the number of possible tic-tac-toe games is at most the number of sequences, each a rearrangement of at most 9 symbols, so at most 9!.

```
1 2 3     . x o
4 5 6     x o .
7 8 9     o x .
```

Give the move sequence (in both full and simpler notation) of a different game that yields the same position. (Different move sequences that yield the same position are called *transpositions* of each other.)

Depending on the amount of game symmetry, saving intermediate results and checking for symmetry can save time. In tic-tac-toe, rotating a position 90 or 180 or 270 degrees, or reflecting the position through a horizontal or vertical or diagonal axis, yields a symmetric position with the same minimax value as the original. Give three different positions symmetric to (and different from) the above position.

---

5. [1 marks] A two-dimensional board position can be represented as a one-dimensional sequence, e.g. row-by-row. With entries 0 (empty), 1 (x) and 2 (o), the above tic-tac-toe position has sequence (. x o x o . o x .) or (0 1 2 1 2 0 2 1 0). A *transposition table* stores results of positions whose transpositions or symmetric equivalents have already been seen. If you solve tic-tac-toe with minimax and a transposition table, explain briefly why the number of entries in the table will be at most 3^9.

---

6. [2 marks] In repo `games-puzzles-algorithms/simple/ttt`, copy `ttt_classic.py` to new program `ttt2.py` and modify `ttt2.py` so that it never calls `non_iso_moves`. So, comment out lines 196 and 249. What additional changes do you need to make to `ttt2.py`?

____

____

How many nodes are examined when solving tic-tac-toe

with `ttt2.py` and no transposition table? ____________

with `ttt2.py` and a transposition table? ____________

with `ttt_classic.py` and no transposition table? ____________

with `ttt_classic.py` and a transposition table? ____________

7. [3 marks]

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

For this tic-tac-toe position with x to play, what is the minimax result: X wins, O wins, or draw? ____________

Below, justify your answer with a proof tree. Hint: `genmove` in `ttt_classic.py`