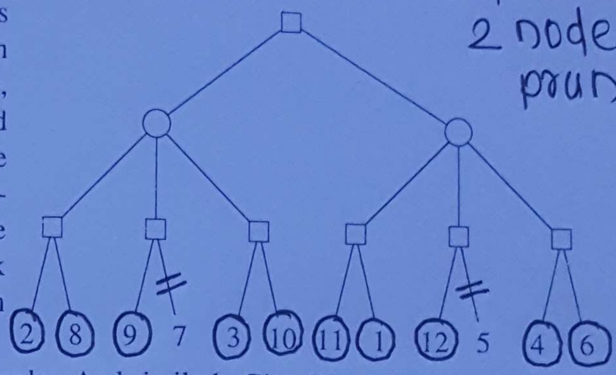
Roll No:

B

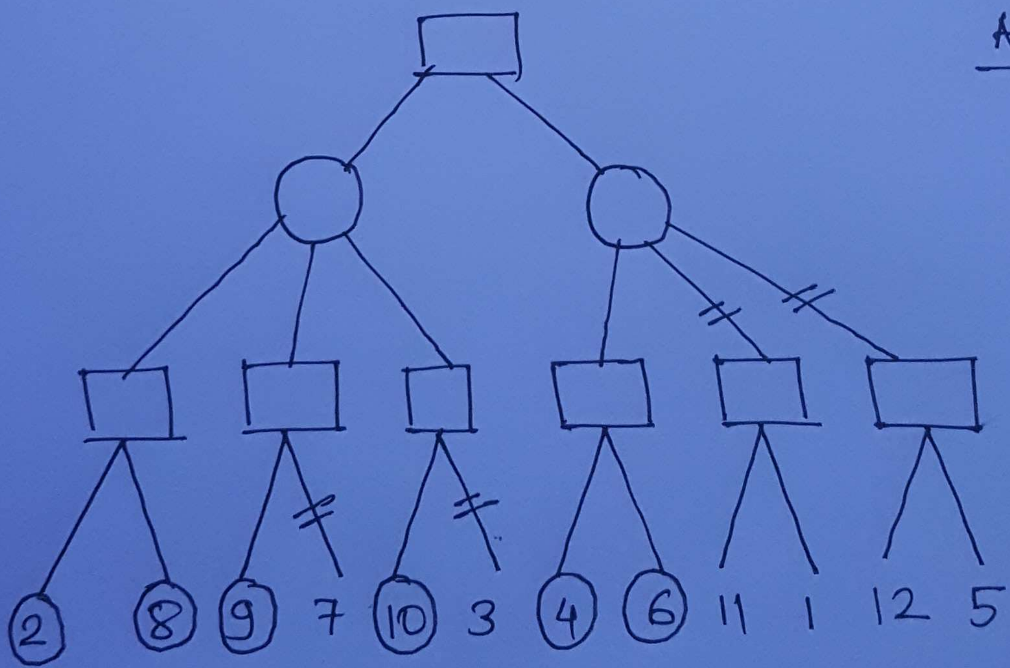
Computer Science and Engineering
IIT Bombay, CS 621 Midsem Exam, Sep 9 2016 15:00 to 17:00
Only 1 A4 page notes allowed. No calculators.
All rough work only on backsides and last page.
Write crisp answers clearly without overwriting for full marks.

1. (24 marks) Sita is playing a game with Gita and it is her turn to move. The game tree is shown on the right for 3-plies (Sita's move, Gita's response, Sita's next move) and the leaf nodes are the values of the game to Sita at that point. If Sita uses alpha-beta search, circle all the leaf nodes she will visit. Work out answer first on back side. Then circle cleanly on this diagram without overwriting.



Ans.
2 nodes pruned.

Suppose Sita can generate her moves in a different order. And similarly Gita. Then the tree will look different since siblings (at any level) can be swapped. For example, the leftmost subtree values can become 8 followed by 2 if Sita generates the moves in this order at that stage. Find such a re-ordered tree (you can only swap sibling nodes at any level) where maximum pruning occurs when Sita does alpha-beta search. Mention how many nodes are pruned and draw tree below circling the leaves evaluated.



Ans.
6 nodes pruned.

Roll No:

B

2. (24 marks) Consider the following set of 6 propositional logic clauses.

$$C = \{q \vee r,\ \neg r \vee \neg v,\ p \vee \neg v,\ \neg p \vee s,\ \neg s \vee \neg r \vee \neg m,\ \neg m \vee r \vee \neg s\}$$

Give one satisfying assignment with least number of propositions having value *true*.

q = true, rest all false (or r = true, rest all false)

Give one satisfying assignment with maximum number of propositions having value *true*.

m, r = false, other 5 vars are true

How many different satisfying assignments are possible?

13 possible solutions (explained in class)

3. (24 marks) Consider formulae $F_1 = (\exists x Q(x)) \Rightarrow (\exists y P(y))$ and $F_2 = \exists x (Q(x) \Rightarrow P(x))$.

(a) Does $F_2 \Rightarrow F_1$? Prove using resolution or give a counter-example using a simple domain with a few constants.

No, $F_2 \not\Rightarrow F_1$

x	$Q(x)$	$P(x)$	F_1	F_2
a	f	f	f	t
b	t	f		

F_2 is true, F_1 is false

(b) Does $F_1 \Rightarrow F_2$? Prove using resolution or give a counter-example using a simple domain with a few constants.

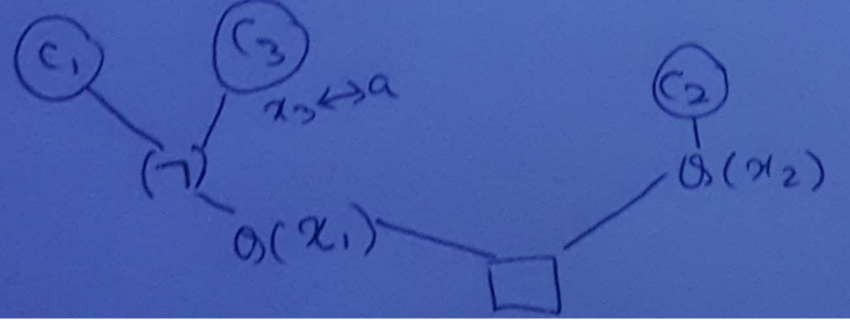
yes, we will prove $F_1 \wedge \neg F_2$ is a contradiction.

Clausal form of F_1 = $\forall x\ \neg Q(x) \vee \exists y\ P(y)$

after scholemization & removing universal quantifiers = $\underbrace{\neg Q(x_1) \vee P(a)}_{C_1}$

clausal form of F_2 = $Q(x) \wedge \neg P(x)$

= $\underbrace{Q(x_2)}_{C_2}$, $\underbrace{\neg P(x_3)}_{C_3}$



Roll No:

B

4. (24 marks) Consider the following Prolog code defining 3 predicates p1, p2 and p3. For every query below assume you repeatedly backtrack by typing ";" after each answer.

```
p1([],[]).
p1([A|R],[B|S]) :- p1(R, S).

p2(A,[],[A]).
p2(A,[B|R],[A| [B | R]]).
p2(A,[B|R],[B|R1]) :- p2(A,R,R1).

p3([],[]).
p3([A|R],[B|S]) :- p1(R, S), p2(A, S1, [B|S]), p3(R,S1).
```

The query `p1([1,2,3],Ans)` will succeed 1 times and give the following values for *Ans* in this order

Ans = [X1, X2, X3]

The query `p2(7, [3,1,4],Ans)` will succeed 4 times and give the following values for *Ans* in this order

[1, 3, 1, 4]
[3, 7, 1, 4]
[3, 1, 7, 4]
[3, 1, 4, 7]

The query `p3([1,2,3],Ans)` will succeed 6 times and give the following values for *Ans* in this order

The query `p3(Ans, [1,2,3])` will succeed 6 times and give the following values for *Ans* in this order