Birla Institute of Technology & Science, Pilani

1st Semester 2023-24

CS F214 - Logic in Computer Science - Mid Sem (Open Book)

==

Time: 90 minutes **Oct 14, 2023** **Marks: 50M**

==

Instructions:

- Answer all the questions in the answer sheet provided.
- **IF YOU HAVE A DOUBT / CLARIFICATION, make your assumption, state it and write. PLEASE DON'T CALL THE INSTRUCTOR.**

Q1. Consider a new set of predicate logic formulas formed over and above the regular predicate logic formulas with the following additions/modifications:

- Introduce the double implication operator (with usual semantics) that has lower precedence than the regular implication operator.
- I wish to enforce my predicate logic formulas only to contain binary predicates.
- I also wish to enforce my predicate logic formulas only to contain unary functions.
- Function terms can be composed only two times i.e. "fun(X)" and "fun(fun(X))" are valid function terms but not "fun(fun(fun(X)))".

(a) Write complete grammar for the new set of predicate logic formulas.

(b) Use the grammar from 1(a) to parse the following predicate logic formula by drawing its parse tree.

$$\forall \boldsymbol{X}\ \forall \boldsymbol{Y}\ \boldsymbol{p1}(\boldsymbol{X},\boldsymbol{Y}) \wedge \left(\exists \boldsymbol{Z}\ \boldsymbol{p2}(\boldsymbol{X},\boldsymbol{Z}) \vee \boldsymbol{p3}(\boldsymbol{Z},\boldsymbol{f1}(\boldsymbol{f2}(\boldsymbol{Y}))\,)\right) \rightarrow \boldsymbol{p1}(\boldsymbol{f1}(\boldsymbol{Z}),\boldsymbol{Z})$$

[Estimated time: 22 mins **8+4=12M]**

Q2 (a) Identify the free and bound occurrences of each of the variables. Illustrate clearly.

$$\forall \mathbf{Y}\ (\mathbf{p}(\mathbf{X}) \wedge\ \mathbf{q}(\mathbf{Y}) \rightarrow \exists \mathbf{X}\ \mathbf{p}(\mathbf{Y}) \vee \mathbf{q}(\mathbf{Z}) \vee \neg \mathbf{r}(\mathbf{X})$$

(b) Perform **[f(A)/X]** on the above formula and write the resultant formula.

[Estimated time: 5 mins **2+2=4M]**

Q3. Consider the following propositional logic formula:

$$\boldsymbol{p_1} \wedge (\boldsymbol{p_4} \rightarrow \boldsymbol{p_2}) \wedge \neg(\boldsymbol{p_1} \wedge \boldsymbol{p_2} \wedge \neg \boldsymbol{p_5}) \wedge \boldsymbol{p_6} \wedge (\boldsymbol{p_6} \rightarrow \boldsymbol{p_4}) \wedge \boldsymbol{p_2}$$

(a) Convert the above formula into a Horn formula

(b) Apply HORN_SAT on the above formula to check for its satisfiability. Write the trace of each iteration.

[Estimated time: 8 mins **2+4=6M]**

Q4. Prove the validity of the following sequent using natural deduction:

$$\left(\forall \boldsymbol{X}\ \exists \boldsymbol{Y}\ \boldsymbol{r}(\boldsymbol{X},\boldsymbol{Y})\right) \rightarrow \left(\neg \exists \boldsymbol{X}\ \boldsymbol{r}(\boldsymbol{X},\boldsymbol{X})\right), \exists \boldsymbol{X}\ \forall \boldsymbol{Y}\ \boldsymbol{r}(\boldsymbol{Y},\boldsymbol{X})\ |-\ \forall \boldsymbol{X} \neg\ \boldsymbol{r}(\boldsymbol{X},\boldsymbol{X})$$

[Estimated time: 22 mins **12M]**

Q5. Given the following predicates:

(a) Let P(x), Q(x), and R(x) be the statements “x is a clear explanation,” “x is satisfactory,” and “x is an excuse,” respectively. Suppose that the domain for x consists of all English text. Express each of the following statements in predicate logic using quantifiers, logical connectives, and P(x), Q(x), and R(x):

(i) All clear explanations are satisfactory and some excuses are unsatisfactory.

(ii) Some excuses are not clear explanations.

(b) Does (i) imply (ii)? Explain briefly.

(c) Does (ii) imply (i)? Explain briefly.

[Estimated time: 8 mins 2+2=4M]

Q6. Write a Prolog predicate **`perms/2`** that generates all permutations of a given list. A permutation is a rearrangement of the elements of the list. For example, given the input:

```
perms([1,2,3], PermList).
```

the predicate should generate **`PermList`** as a list of all possible permutations of **`[1,2,3]`**, such as

```
PermList = [1,2,3] ;
PermList = [1,3,2] ;
PermList = [2,1,3] ;
PermList = [2,3,1] ;
PermList = [3,1,2] ;
PermList = [3,2,1] ;
false.
```

You can create any other predicate(s) as helper predicate(s) to define **`perms`**.

[Estimated time: 22 mins 12M]

====== End of the Document ======