INDIAN INSTITUTE OF TECHNOLOGY ROORKEE
Department of Computer Science and Engineering
Spring Semester 2018-19 End Term Examination 24.4.2019
CS312 Principles of Programming Languages FM: 50 Duration: 3hour

Answer **all the** questions. Answers should appear in **the order** of the questions.
Answer to each question **must begin** in a **new page.**
Parts of the same question should appear in a contiguous manner.
No mark will be given **if multiple answers are written** for a question.
Answers **written in PENCIL shall not be evaluated. Use PEN for diagrams.**
Answers that **violate** the above instructions will be awarded **zero** mark.

DO NOT WRITE ANYTHING ON THE QUESTION PAPER

1. (i) Define a lifted set for a given set X.
(ii) Define an information ordering relation on a lifted set.
(iii) Define an information ordering relation on the Cartesian product of two lifted sets.
(iv) Using the Definition in 1.(iii), construct the mathematical object when X = {true, false}. State the name of the mathematical object. [2+2+2+4]

2. (i) State and briefly justify the rules for **if-then-else**, **while**, and **assignment** statements for proving partial correctness of computer programs.
(ii) Establish the partial correctness of the following program by giving a suitable loop invariant. Justify the choice of the loop invariant.
The precondition is **y ≥ 0**. The post condition is **z = y × c**. [5+5]

```
a := 0;
z := 0;
while ( a ≠ y)
{
      z := z + c;
      a := a + 1;
}
```

3. (i) Compute the normal form of M N, where M = (λx. x x) and N = (λy. λz. y z). Justify all the steps clearly.
(ii) Design simple lambda terms for the Boolean operations AND and XOR. Explain how you arrived at the result. Merely stating the lambda terms shall yield zero mark. [5+5]

4. (i) Give a recursive definition of **add** function for addition of two numbers m, n. Show the computation of **add** in lambda calculus for $m = 1, n = 2$. Assume call-by-name semantics.
(ii) Find the principal type of the term (λx. λy. λz. x y z z) using the Principal Type algorithm. Show all the steps. [5+5]

5. (i) Construct a derivation tree for the term given below and use it to infer the type of the term (if any).
if iszero (succ 0) then succ(succ 0) else pred(succ 0)
(ii) Give appropriate mathematical representations of the following and briefly explain:
(a) The function **age: employee → int** can be used for all subtypes of **employee**.
(b) A class that has a list of elements as data and only one function that returns the last element of the list. [5+5]

****END***