Computer Science & Engineering 505 – Midterm

November 1, 1999

Open book & notes – 50 minutes – 10 points per question

50 points total

Name: ______________________________

Please write any long answers on the back of the exam or on a separate piece of paper.

1. Consider the following program in an Algol-like language.

```
begin
integer g;
procedure clam(j,k: integer);
    begin
    print(j,k);
    g := g+1;
    print(j,k);
    end;
g := 1;
clam(2*g, g);
end;
```

   What is the output if `j` and `k` are both passed by:

   (a) call by value

   (b) call by name

   (c) call by reference

   (Assume that having an expression as the actual parameter when passing by reference is legal. The compiler generates a temporary for the result of evaluating the expression, which is then passed by reference.)

2. Suppose that you write a random number generator in Algol-60, and use an `own` variable in the procedure `random` to hold the seed. Suppose also you want to be able to initialize the seed to a given value, so that the random number generator will generate the same values for each run of the program for testing purposes. How would you initialize the seed? Discuss the concept of `own` variables in light of your answer. If they worked well for this usage, say so; if they didn't work well, discuss alternatives.

3. Consider the following linear programming problem.

   minimize x
   subject to
   $4 \leq x$
   $x \leq 10$

   As usual, x is constrained to be non-negative.

   We first eliminate the inequality constraints by adding slack variables s_1 and s_2:

   $4 + s_1 = x$
   $x + s_2 = 10$

   After the first phase of the simplex algorithm, we might obtain the following basic feasible solution:

   $s_1 = 6 - s_2$
   $x = 10 - s_2$

   (a) What is the solution given by this tableau?

   (b) What is the value of the objective function?

   (c) Starting from this tableau, show how you obtain an optimal solution.

4. Our implementation of CLP($\mathcal{R}$) uses an incomplete solver. Call this one "Standard CLP($\mathcal{R}$)". Suppose that we had another implementation, called "Complete CLP($\mathcal{R}$)", that has a complete solver. Both Standard and Complete CLP($\mathcal{R}$) try rules in the same order, and select the leftmost literal in a derivation.

   (a) Are there goals for which Standard CLP($\mathcal{R}$) returns an answer, and for which Complete CLP($\mathcal{R}$) says no? Why? Give an example if one exists.

   (b) Are there goals for which Complete CLP($\mathcal{R}$) returns an answer, and for which Standard CLP($\mathcal{R}$) says no? Why? Give an example if one exists.

   (c) Suppose both Standard CLP($\mathcal{R}$) and Complete CLP($\mathcal{R}$) return an answer to a given goal (i.e. the final constraint store simplified with respect to the variables in the initial goal). Are these answers always equivalent with respect to the variables in the initial goal? Or are there cases in which they are not equivalent but one implies the other? Are there cases in which neither implies the other? Give examples.

5. Consider the following CLP($\mathcal{R}$) rules.

```
smallprime(2).
smallprime(3).
smallprime(5).
```

Show the derivation tree for the following goal.

```
smallprime(N), N>2.
```