CSE 341 Final — June 8, 2000

Your name:

This exam is open book, open notes, 110 minutes, 110 points total.

1. (18 points) Consider the following program. This program will print out six values.

```
begin
integer n;
procedure p(j: integer)
    begin
    n := 100;
    j := j+10;
    print(n);
    print(j);
    end;
n := 0;
p(n);  print(n);
p(n);  print(n);
end;
```

   (a) What is the output if the parameters are passed by value?

   (b) What is the output if the parameters are passed by value-result?

   (c) What is the output if the parameters are passed by reference?

2. (5 points) A static method in Java may not use the `this` reference. Why not?

3. (27 points) Suppose that the following Miranda script has been filed in.

```
triple x = 3*x
add1 x = x+1

makebit False = 0
makebit True = 1

my_const k x = k

function_map [] x = []
function_map (f:fs) x = f x : function_map fs x

tree * ::= Leaf * | Node (tree *) (tree *)

height (Leaf x) = 0
|| max2 is a library function that finds the max of two numbers
height (Node left right) = 1 + max2 (height left) (height right)

treesum (Leaf x) = x
treesum (Node left right) = treesum left + treesum right

mytree = Node (Leaf 10) (Node (Leaf 20) (Leaf 30))
```

What is the result of evaluating the following Miranda expressions? If there is a compile-time type error, or a run-time error, or a non-terminating computation, say so. If the result is infinite, give the first several values. If the expression is followed by `::`, then give the type, instead of the value.

(a) `my_const ::`

(b) `function_map ::`

(c) `function_map [triple, makebit] ::`

(d) `function_map [triple,add1, my_const 100 ] 10`

(e) `function_map [my_const 100 ] ::`

(f) `function_map [my_const 100, makebit] ::`

(g) `height ::`

(h) `treesum ::`

(i) `function_map [height,treesum] mytree`

4. (10 points) Consider the following definition of the `min` constraints in CLP($\mathcal{R}$).

```
min(X,Y,X) :- X<=Y.
min(X,Y,Y) :- X>Y.
```

Show the derivation tree for the following queries. (You can show either the full or simplified tree, or something in between.)

(a) `min(5,A,2).`

(b) `min(5,A,10).`

5. (12 points) Consider the following definition of the `min` and `sorted` constraints in CLP($\mathcal{R}$). (`min` is the same as in Question 4.)

```
min(X,Y,X) :- X<=Y.
min(X,Y,Y) :- X>Y.

sorted([]).
sorted([X]).
sorted([A,B|R]) :-
  A <= B,
  sorted([B|R]).
```

What answers are produced for the following queries? (Give all of the answers that would be produced by backtracking.)

(a) `min(A,B,10).`

(b) `sorted([20,A,10]).`

(c) `sorted([A,B]), min(A,B,20).`

(d) `sorted([A,B,C,D]), sorted([D,A,B,C]).`

6. (18 points) Consider the following Java class definition.

```
class PairOfStrings {

  public String a, b;

  public PairOfStrings(String a, b)
    {this.a = a; this.b = b;}

  public void flip()
    {String temp; temp = a; a = b; b = temp;}

  public boolean equals (PairOfStrings p)
    {return (a.equals(p.a) && b.equals(p.b));}

}
```

What is printed by the following code?

```
PairOfStrings p1 = new PairOfStrings("clam","squid");
PairOfStrings p2 = p1;
PairOfStrings p3;


System.out.println(p1 == p2);

System.out.println(p1.equals(p2));


p2.flip();

System.out.println(p1 == p2);

System.out.println(p1.equals(p2));


p3 = new PairOfStrings("squid","clam");

System.out.println(p1 == p3);

System.out.println(p1.equals(p3));
```

7. (10 points) Suppose that type checking (both at compile time and at run time) was eliminated entirely from Miranda. Are there any Miranda functions that currently don't compile due to a type error but that would run correctly in the version of the language with no type checking? If so, give an example. If not, explain why there aren't any such functions.

8. (10 points) Describe two significant similarities between Java and Smalltalk, and two significant differences.