Spring, 1993 Name: ______________________

Com S 541 — Programming Languages 1

Test 1 (Concepts)

This test is open book and open notes.
Before you begin, please take a moment to look over the entire test so that you can budget your time.
Short answer questions will be graded on the basis of completeness (did you list all the relevant parts of the answer?) soundness (did you make any errors of fact or contradict yourself?) and clarity. You can help ensure clarity by giving examples where possible.
For essays, critical justifications, and the like there may be no one right answer; so the criteria for judgement will be whether your essay is:

- clear (you use examples where appropriate, you write clear sentences, diagrams, and you avoid excess verbosity),
- sound (you start from facts or reasonable assumptions, your logic is convincing), and
- complete (you consider relevant aspects of the problem and alternatives).

The highest grade will be earned by an answer that is, in addition, concise and elegant; that is, answers that say the most with the least amount of excess. Extra consideration will be given to answers that are especially creative.

1. (10 points) Which is better: name equivalence or structural type equivalence or both? Briefly justify your answer. Give examples if that helps clarify your reasoning.

2. (10 points) The language ML provides two notations for cartesian product types.

```
type employee1 = string * string * real;

type employee2 =  {familyname : string,
                   givenname  : string,
                   salary     : real    };
```

   Are these notations equivalent in programming? Why or why not? Are the types `employee1` and `employee2` semantically equivalent? Why or why not?

Do *one* of the following 2 problems.

3. (5 points) In C and C++ the operator `&` allows a programmer to take the address of an arbitrary variable. What trouble can be caused by careless use of the `&` operator in C or C++ programs? Give an example.

4. (5 points) What is the type that ML would infer for the following function?

```
fun K(x) =
  (fn y => x);
```

(The notation `fn y => x` is a function that takes an argument y, and returns the value of `x`.)

5. (10 points) Consider the following program.

```
PROGRAM test

  CONST z = 10;
  VAR a: integer;  b: integer;

  PROCEDURE test(x: integer, y: integer)
   BEGIN
     writeln("a =", a, " b =", b, " x =", x, " y =", y);
     x := z;
     y := x * a;
     b := y + a;
     writeln("a =", a, " b =", b, " x =", x, " y =", y)
   END;

BEGIN
  a := -5;
  b := 3;
  test(a, b);
  writeln("a =", a, " b =", b)
END
```

The procedure `writeln` writes its arguments on a single line. Give the output of the above program, assuming that all parameters are passed using the mechanisms of:

(a) call by value-result, and

(b) call by reference.

6. (25 points)

(a) Pick a language that you are familiar with from the following list: FORTRAN, C, C++, Pascal, LISP, Scheme, Standard ML, Modula-2, or Ada. Cirle the name of your choice.

(b) List 3 (or more) features of the language that you picked that you believe are most in need of improvement. (These features should have something to do with what has been discussed in the book, readings, homework, or in class.) Briefly describe the problems with the each feature.

(c) Rank the 3 features, and explain which is most in need of improvement, which is next most important, and which is least important. Briefly justify your rankings.

(d) Describe, with as much precision as you can, how you would improve (fix) the feature you ranked as most in need of improvement. Briefly justify why your changes are an improvement.