UCLA Computer Science 131 (Spring 2008)
Midterm
100 minutes total, open book, open notes

Name:______________________ Student ID:__________

1	2	3	4	5	6	7	total

1 (5 minutes). "Ireland has leprechauns galore." is an example of a particular kind of syntactic construct in English. Can you construct a similar example in C++, OCaml, or Java? If so, give an example; if not, explain why not.

2a (7 minutes). Write an OCaml function 'twice' that accepts a function f and returns a function g such that g(x) equals f(f(x)). For simplicity's sake, you can assume that f is free of side effects, and you can impose other restrictions on f and x. Try to keep the restrictions as minor as possible, and explain any restrictions you impose. Or, if 'twice' cannot be written easily OCaml, explain why not.

2b (7 minutes). Same as (a), except write a function 'half' that accepts a function f and returns a function g such that f(x) equals g(g(x)).

2c (6 minutes). Give the types of 'twice' and 'half'.

3. Consider the following grammar for a subset of the C++ language.

```
   expression:
     expression ? expression : expression
     expression != expression
     expression + expression
     ! expression
     INTEGER-CONSTANT
     ( expression )
```

For example, (!!0+1!=2 ? 3 : 4) is read as "if not-not-0 plus 1 does not equal 2, then 3, else 4", and evaluates to 4.

3a (2 minutes): What are the tokens of this subset of C++?

3b (5 minutes): Show that this grammar is ambiguous.

3c (10 minutes): Rewrite the grammar so that it is no longer ambiguous, resolving any ambiguities in the same way that C++ does. Recall that in

C++, the expression

```
(0 != 1 != 2 || 3 + !4 + 5 || 6 ? 7 : 8 ? 9 : 10)
```

is treated like

```
(((((0 != 1) != 2) || ((3 + (!4)) + 5)) || 6)
 ? 7 : (8 ? 9 : 10))
```

3d (8 minutes): Translate the rewritten grammar into a syntax diagram.

4 (10 minutes). A numerical analyst is really bothered by the special values of IEEE floating point, and asks you to modify Java to fix what she views as a serious conceptual flaw. She wants her Java programs to throw an exception instead of returning infinities and NaNs. Is her request reasonable for Java programs? Is it implementable? Why or why not? Don't worry about compatibility with existing compilers, etc.; assume that you are the inventor of Java and she is asking for this feature early in your language design process.

5 (5 minutes). Give an example of four distinct Java types A, B, C, D such that A is a subtype of B, A is a subtype of C, B is a subtype of D, and C is a subtype of D. Or, if such an example is impossible, explain why not.

6 (10 minutes). Explain how you would implement OCaml-style type checking, in an implementation that uses dynamic linking heavily. What problems do you foresee in programs that relink themselves on the fly?

7a (10 minutes). Write a curried OCaml function "interleave C S L1 L2" that constructs a new list L from the lists L1 and L2, using the chooser C with seed S, and returns a pair (S1, L), where S1 is the resulting seed and L is the interleaved list. For example, "interleave C S [1;2] [3;4;5]" might invoke C four times and then return (S1, [1;3;4;2;5]). At each step of the iteration, "interleave" should use the chooser to decide whether to choose the first item of L1, or the first item of L2, when deciding which of the two items to put next into L. If L1 is empty, the chooser need not be used, since "interleave" will just return L2; and similarly if L2 is empty, "interleave" should just return L1 without invoking C.

For the purpose of this question, a "chooser" is as defined in Homework 1.

"interleave" should invoke C a minimal number of times, left to right across the lists L1 and L2. "interleave" should avoid side effects; it should be written in a functional style, without using

OCaml libraries.

7b (10 minutes): Write a function "outerleave" that does the opposite of what "interleave does": it splits a list into two sublists that can be interleaved to get the original list, and returns a triplet consisting of the new seed and the two sublists. That is, "outerleave C S [1;3;4;2;5]" might yield (S1, [1;3;2], [4;5]). If given a list of length N, "outerleave" always invokes the chooser N times.

7c (5 minutes): Give the data types of all top-level values or functions defined in your answer to (a) and (b). If you did not finish (a) and (b), give the best answer you can, and state your assumptions about what a solution would look like.