CS 342 — Principles of Programming Languages October 6, 1989

Test

Please write your name here:

Before you begin, please take a moment to look over the entire test so that you can budget your time.

Answers to essay questions will be judged on their completeness and clarity. However, please do not feel obligated to fill up all the empty space on the page. Conversely, do not feel limited by the space on a page, since you may continue an answer on the back of a page if you wish.

problem	subject	points	score
1	costs and benefits	27	
2	language comparison	12	
3	FORTRAN COMMON	10	
4	pseudo-code implementation	10	
5	Algol 60 call by name	25	
6	Syntax	16	
Total		100	

This exam (only) counts for 15% of your final grade.

1. Describe the significant differences in costs (or benefits) between the two features listed in each of the following pairs.

   (a) (9 points) call by reference and call by value

   (b) (9 points) structural type equivalence and name type equivalence

   (c) (9 points) static scoping and dynamic scoping

2. (12 points) Why is FORTRAN a better language than the pseudo-code language that you have implemented?

3. (10 points) Describe the problems caused by FORTRAN's COMMON facility.

4. (10 points) Describe the main data structures used in your implementations of the pseudo-code interpreter and how they are used in the read-execute cycle.

5. (25 points) Give the output of the following Algol 60 program (note that the parameters to `s` are passed by name). Assume that the `printInteger` routine prints its argument on a separate line of output.

```
begin
        integer i,j;
        integer array A[0:2];

        procedure s(j,i);
                integer j,i;
                begin
                        for i := 0 step 1 until 2 do
                                j := j × j
                end;

        for i := 0,1,2 do
                begin A[i] := i end;
        i := 1; j := 0;
        s(A[j],j);
        printInteger(i); printInteger(j);
        for i := 0,1,2 do
                begin printInteger(A[i]) end;
end
```

6. Consider the following grammar.

$\langle$`E`$\rangle$ ::= $\langle$`bool exp`$\rangle$ | $\langle$`int exp`$\rangle$
$\langle$`bool exp`$\rangle$ ::= $\langle$`bool var`$\rangle$ | $\neg$ $\langle$`bool exp`$\rangle$
$\langle$`bool var`$\rangle$ ::= $\langle$`id`$\rangle$
$\langle$`int exp`$\rangle$ ::= $\langle$`int var`$\rangle$ | $\langle$`E`$\rangle$ `[` $\langle$`int exp`$\rangle$ `]`
$\langle$`int var`$\rangle$ ::= $\langle$`id`$\rangle$
$\langle$`id`$\rangle$ ::= `x` | `y` | `z`

(a) (2 points) Write two different sentences in the language generated by the above grammar.

i.

ii.

(b) (8 points) Prove that the above grammar is ambiguous.

(c) (6 points) Write a regular expression for the strings generated by the nonterminal $\langle$bool exp$\rangle$. (That is, if $\langle$bool exp$\rangle$ was the start symbol of the above grammar instead of $\langle$E$\rangle$, what regular expression would describe that language?)