Com Sci 221 — Programming Languages
Midterm Exam — Winter 1991
Due Wednesday, February 20
2 pages of problems, 7 problems
Open book, open notes, open mind

Take this exam at a time and place of your own choosing, but in time to hand it in at the beginning of class on Wednesday, February 20. Give yourself a contiguous 2-hour block of time to do the problems, starting when you open the exam paper. You may refer to your text, notes or any other written materials gathered in advance. You may not communicate with anyone else about the exam or class material during your exam period. Each numbered problem is equally valuable.

1. Describe, in English or in RAM code, the execution of the array assignment command A[A[i]]:=A[j]. Be careful about the order of evaluating the various l-values and r-values. Give an example of values for A, i, and j for which A[A[i]]$\neq$A[j] immediately after execution of the command.

2. Recall that $A \times B$ is the set of ordered pairs (records) with a component from A and a component from B; $\mathcal{P}(C)$ is the set of subsets of C (Pascal's *set of* C). Discuss briefly (approximately 1 page) the relation between the set-theoretic types $C = \mathcal{P}(A \times B)$ and $D = \mathcal{P}(A) \times \mathcal{P}(B)$. In particular, does either one of C or D represent in a natural way all of the members of the other? Suppose that A is the set of students at the University of Chicago, and B is the set of courses offered. Include in your discussion an example of the sort of information that could be represented by members of C, and an example of the sort of information that could be represented by members of D.

3. $A \rightarrow B$ denotes the set of functions from A to B (*array* A *of* B), and $A + B$ denotes the marked union of A and B (variant record containing either a member of A or a member of B). The following 6 types form 3 pairs of essentially equivalent types. What are the pairs?

   (a) $A \rightarrow (B \rightarrow C)$

   (b) $A \rightarrow (B \times C)$

   (c) $(A \rightarrow C) \times (B \rightarrow C)$

   (d) $(A \rightarrow B) \times (A \rightarrow C)$

   (e) $(A \times B) \rightarrow C$

   (f) $(A + B) \rightarrow C$

4. Draw the abstract syntax tree for the expression

$$a + ((b + (c + d)) + e)$$

   Translate the expression into postfix notation, and give straightforward machine code to evaluate the expression. Present the postfix and the machine code side by side, and show the simple connection between them.

5. Draw the abstract syntax tree and the flowchart for the schematic program fragment

α;
while A **do begin**
  β;
  if B **then begin**
    γ
  end else begin
    δ
  end
end

Draw a box around the portion of the flowchart that corresponds to each subtree in the abstract syntax tree. What nice property of the contents of these boxes do you observe, that is guaranteed to hold for all flowcharts derived from "structured" programs.

6. The Fibonacci numbers are defined by the recursion:

$$F(0) = 0$$
$$F(1) = 1$$
$$F(n+2) = F(n) + F(n+1)$$

Consider the following program fragment that computes the nth Fibonacci number:

$a, b := 1, 0$;
$i := 1$;

while $i < n$ **do begin**
  $a, b := a + b, a$;
  $i := i + 1$
end

Give appropriate precondition, postcondition, and loop invariant to show that the program fragment is correct (they are *very* simple).

7. Give a very brief and simple example of a single procedure declaration and a call to the procedure that yields 3 different results using call by value-result, call by reference, and call by name. The actual parameters in the procedure call must all represent legitimate l-values, and the call must terminate normally for all three parameter-passing methods.