CSCE 330 Fall 2003
MIDTERM EXAM
Thursday 2003-09-25—Three Pages, Closed Book

1 Short Questions–1 or 2 points each; 15 points total

1. (1 point) What is the name of the family of programming languages whose structure is dictated by the von Neumann computer architecture? **Answer:** Imperative
2. (2 points) Following up on the previous question, what are the names of the two other main families of programming languages? **Answer:** functional (or applicative) and logic
3. (1 point) Consider the four programming languages Fortran, Algol 60, LISP, and Pascal. Which one of them was designed by a committee? **Answer:** Algol 60
4. (1 point) Match: (a) Verification and (b) Validation with (1) "Are we building the program right?" and (2) "Are we building the right product?" **Answer:** a-1, b-2
5. (1 point) What does the acronym RUDE stand for? **Answer:** Run Understand Debug Edit
6. (1 point) The spiral method of software development is characterized by prototyping. True or false? **Answer:** True
7. (1 point) Match 1–2 with 3–4 and with 5–6:
   (a) Syntax
   (b) Semantics
   (c) Meaning
   (d) Form
   (e) How something is expressed
   (f) What something does

   Answer: Syntax-Form-How Someting Is Expressed; Semantics-Meaning-What Something Does
8. (1 point) Syntax diagrams are a notational variant of EBNF. True or false? **Answer**: True

9. (1 point) What readability problem is caused by using the same reserved word or symbol to close more than one control statement? **Answer:** The reader must match the closing reserved word to the opening reserved word by counting, rather than by matching a token.

10. (1 point) The following sentence is syntactically correct: "Time flies like green bananas." True or false? **Answer:** True.

11. (2 point) What are the three components of the state in the denotational semantics approach[1]? **Answer:** mem, i, o

12. (2 points) There are three major approaches to describing the semantics of programming languages. List them. **Answer:** axiomatic, denotational, operational.

2 Syntax–14 points

1. (2 points) What does it mean for a (context-free) grammar to be *ambiguous*? **Answer:** The grammar generates a sentence that has two or more distinct parse trees.

2. (8 points) The grammar of the original definition of Algol 60 contained the following production rules:
```
<statement> -> <conditional-statement> | begin <statement> end
<conditional-statement> -> if <condition> then <statement>
| if <condition> then <statement> else <statement>
```
Show that any grammar containing these production rules is ambiguous. **Answer:** Provide two distinct parse trees of `if <condition> then if <condition> then <statement> else <statement>`

3. (4 points) Draw a syntax graph corresponding to the following EBNF rule (where the parentheses are metalevel symbols that indicate grouping):
```
<ident> ::= letter (letter | digit)*
```

3 Semantics–20 points

1. (3 points) Recall that, in denotational semantics, the *state* is a triple `<mem, i, o>`.

   (a) Which component(s) of the state are changed by a *read* instruction? **Answer:** mem, i

   (b) Which component(s) of the state are changed by a *write* instruction? **Answer:** o

   (c) Which component(s) of the state are changed by an assignment? **Answer:** mem

[1] Every question on denotational semantics refers to the simple language described in class.

2. (2 points) Consider the simple language described in class in the lectures on denotational semantics. Assuming the absence of errors, complete the denotational semantics of the while-do-od loop, by identifying what *mystery* is.

   $dsem_{DO}$(while B do SL od, s) = *mystery* if $dsem_{BOOL}$(B, s) = false, otherwise
   $dsem_{DO}$(while B do SL od, s) = $dsem_{DO}$(while B do SL od, $dsem_{DO}$(SL, s))

   Answer: s

3. (5 points) Suppose that you have a `while` loop `while B do SL end`, where B is a Boolean expression and SL is a statement list. Suppose further that you have showed that an assertion I holds before the `while` and that I is invariant upon execution of SL, i.e. $I \wedge B \Rightarrow I$. You want to show that upon termination of the `while`, the assertion Q holds. What do you need to prove? **Answer:** $I \wedge \neg B \rightarrow Q$.

4. (5 points) Describe (very briefly) the semantic difference between commands and expressions. (Hint: Use the assignment statement, which is a command, to illustrate the difference.) **Answer:** commands change the state, expressions yield a value.

5. (5 points) Give a loop invariant for this loop:

```
while (i <> n) do
  begin
    sum := sum+i;
    i := i+1
  end
```

   with precondition $i = 1 \wedge sum = 0$ and postcondition $sum = \sum_{j=1}^{n-1} j$.

   Answer: $sum = \sum_{j=1}^{i-1} j$