Midterm Test Solutions (Evening section)

October 31, 2002

Duration: 50 minutes

Aids allowed: None

Weight: 20% of your course grade

This exam contains a total of 5 pages (including this one). Write your answers clearly in the spaces provided. Use the back pages for your rough work.

Surname: __________________________  # 0: _____/ 1

First name: __________________________  # 1: _____/ 9

Student #: __________________________  # 2: _____/ 4

# 3: _____/ 7

# 4: _____/ 5

# 5: _____/ 4

# 6: _____/10

TOTAL: _____/40

Good Luck!
Question 0. [1 mark]
Write your student number legibly in the space provided on every page of this test.

Question 1. Grammars [9 marks]

Part (a) [2 marks]
Consider the following grammar:
\[
\begin{align*}
<S> & \rightarrow <S> \text{ op1 } <G> | <G> \\
<G> & \rightarrow <Z> \text{ op2 } <G> | <Z> \\
<Z> & \rightarrow <Z> \text{ op3 } <M> | <Z> \text{ op4 } <Z> | <M> \\
<M> & \rightarrow x | y \ ( <S> )
\end{align*}
\]

Give the associativity (left, right or neither) or the operators op1, op2, op3 and op4.

<table>
<thead>
<tr>
<th>Operator</th>
<th>Associativity</th>
<th>Operator</th>
<th>Associativity</th>
</tr>
</thead>
<tbody>
<tr>
<td>op1</td>
<td>Right</td>
<td>op3</td>
<td>Left</td>
</tr>
<tr>
<td>op2</td>
<td>Left</td>
<td>op4</td>
<td>Neither</td>
</tr>
</tbody>
</table>

Part (b) [4 marks]
Consider the above grammar again. Which of the following strings are ambiguous? Mark the blank next to each string with either yes if it is ambiguous, or no if it is not ambiguous.

<table>
<thead>
<tr>
<th>Ambiguous?</th>
<th>Ambiguous?</th>
</tr>
</thead>
<tbody>
<tr>
<td>i. <strong>No</strong>__ x op3 x op4 x</td>
<td>iii. <strong>Yes</strong>__ x op4 x op3 x</td>
</tr>
<tr>
<td>ii. <strong>No</strong>__ x op4 x op2 x</td>
<td>iv. <strong>Yes</strong>__ x op4 x op4 x</td>
</tr>
</tbody>
</table>

Part (c) [3 marks]
Write a CFG in BNF for the language of balanced parentheses. For example, your grammar should accept (())() and (((()))(())) but reject () and ()(). Make sure your grammar is unambiguous.

\[
<S> \rightarrow ( <S> ) <S> | \epsilon
\]
Question 2. List Representation [4 marks]

Draw a diagram showing Scheme's internal representation for the following expression:

$$(3\ 4\ ())\ 5\ (7\ 8)\ .\ 9)$$

Question 3. Scheme Procedure [7 marks]

Write a Scheme procedure (insert-sorted x lst) which returns a copy of the sorted list of numbers lst with x inserted in the correct location, such that the returned list is still in sorted in increasing numerical order. If x is already in lst, do not re-insert it. For example,

(insert-sorted 3 ' (1 2 4 5)) returns (1 2 3 4 5).
(insert-sorted 3 ' (1 2 3 4)) returns (1 2 3 4).

(define insert-sorted
  (lambda (x lst)
    (cond ((null? lst) (list x))
          ((< x (car lst)) (cons x lst))
          ((= x (car lst)) lst)
          (else (cons (car lst) (insert-sorted x (cdr lst))))))
  )
)
Question 4.  High-Order Procedures  [5 MARKS]

Write a procedure (count-gt x lst) which returns the number of elements in lst which are
greater than x. Use map and apply to write your procedure. You may not use recursion or eval.
For example,
(count-gt 5 '(4 5 6 7)) returns 2.

; Pre:  x is a number, lst is a list of numbers.
(define count-gt
  (lambda (x lst)
    (apply + (map (lambda (y) (if (< x y) 1 0)) lst)))
  )
)

Question 5.  Parameter Passing  [4 MARKS]

Consider the following code in some theoretical language:

int i, j, k;

subprogram p(int a, int b) {
  a := a + 2;
  b := b - 1;
  i := 6;
}

i := 10;
j := 11;
k := 12;
p(j, k);
print i, j, k;  --- print 1

p(j, i);  --- Note:  p(j, i), not p(i, j)!
print i, j;   --- print 2

What is the output of this program if the parameter-passing mode of this language is pass by
reference?

print 1: i: 6  j: 13  k: 11

print 2: i: 6  j: 15
Question 6. Short Answers [10 marks]

Part (a) [1 mark]
Give one advantage of assembly over machine language.

Easier to write because there are mnemonic for commands, names for variables, and you can use macros. Just writing “Easier to write” got half marks.

Part (b) [2 marks]
Give two advantages of high-level languages over assembly.

Some potential answer: Easier to port; more constructs make abstractions easier; easier to read and write; easier to debug.

Part (c) [1 mark]
Name a key operation of the object oriented programming paradigm.

Inheritance; message passing.

Part (d) [2 marks]
With an ambiguous grammar, how many parse trees are there for any given sentence that is in the language? Circle the best answer:

0 exactly 1 more than 1 1 or more

Part (e) [2 marks]
With an unambiguous grammar, how many parse trees are there for any given sentence that is not in the language? Circle the best answer:

0 exactly 1 more than 1 1 or more

Part (f) [2 marks]
What does (list (car '((1) 2)) (cdr '(3 (4)))) return?

((1) ((4)))

Total Marks = 40