FINAL EXAMINATION, April 1995

CSC190S - Computer Algorithms, Data Structures, and Languages

1. (a) [2 marks] A tertiary (degree 3) tree has 80 nodes. If the tree is of minimum height, how many leaves does it have?

1. (b) [2 marks] A sort is performed as follows. Randomly ordered elements are read from input, and inserted one by one into an initially empty binary search tree. The insertions are done without balancing. An inorder traversal (left subtree, node, right subtree) is then performed to output the nodes in their sorted order. What, on average, is the overall complexity of this sorting algorithm?

1. (c) [2 marks] A binary search tree is built from 14 distinct keys. If the insertion order can be chosen arbitrarily, how many distinct minimum height trees can be formed?

1. (d) [2 marks] In the context of a compiler, what is a "token"?

1. (e) [2 marks] For a given grammar, a sentence is found whose left canonical derivation and right canonical derivation yield different sequences of sentential forms. What may one conclude about the grammar?

1. (f) [2 marks] What is the worst-case time complexity (in 'big O' notation) of data retrieval using a hash table holding n data items, given that the table is never *completely* full?

1. (g) [3 marks] What is the main function of the symbol table in a compiler?

2. (a) [6 marks] The following program segment is written in a generic programming language. Assume that each execution of the procedure "mumble" takes a constant 5 units of CPU time, and that each execution of every other line of code takes 1 unit of CPU time. Develop an expression T(N) for the running time.

```
for i := 1 to N                    line 1
   if i < 10 then                  line 2
      for j := 1 to i              line 3
         call mumble(j)            line 4
      end for                      line 5
   else                            line 6
      call mumble(i)               line 7
   end if                          line 8
end for                            line 9
```

2. (b) [2 marks] What is the complexity (in 'big O' notation) of the program in 2(a) above?

2. (c) [7 marks] Suppose you are asked to suggest an algorithm for sorting data contained in a one-dimensional array (a vector). There will be between 10,000 and 100,000 data items to sort, and it is known that the data is initially *almost* in sorted order. Furthermore, those elements that are out of order are relatively close to their final positions. Describe clearly the algorithm would you propose, and explain its merit. If you feel you need further information, state those questions you would want to ask, and explain how this additional information would affect your recommendation?

3. (a) [2 marks] Consider the set of binary search trees, each of which is balanced and contains 20 nodes. What is the height of the shortest tree in this set?

3. (b) [2 marks] For the same set of trees as in 3(a), what is the height of the tallest tree in this set?

3. (c) [3 marks] For the same set of trees as in 3(a), what is the first possible level at which a leaf will be found? [Follow the convention that the root is at level 1, its children are at level 2, etc.]

3. (d) [6 marks] Divide and conquer has been demonstrated to be an efficient strategy in many problems, such as the quicksort program. The following code, written in a generic language, is designed to find the maximum element in an array using divide and conquer.

```
/* array "A" is global variable */

function findmax(lo, hi)
   if lo >= hi then
      result A[lo]
   else
      /* use divide and conquer to find max in 2 groups */
      mid := (lo + hi) div 2
      max1 := findmax(lo, mid)
      max2 := findmax(mid, hi)
      if max1 > max2 then
         result max1
      else
         result max2
      end if
   end if
end function
```

Does this algorithm work? If it does, comment on its efficiency. If it does not, explain the flaw.

4. [12 marks] A homemaker has two old measuring cups. When filled to the brim, one holds 3 pints, and the other holds 5 pints. The markings have worn off the cups, so it is only when filled to the brim that the amount of liquid in the cup is known accurately. However, pouring between the cups can produce other quantities. For example, if the homemaker wanted to measure out 2 pints, this could be done by filling the 5 pint cup, then pouring this into the 3 pint cup, leaving 2 pints of water in the 5 pint cup.

Initially, both cups are empty. There are 6 possible operations:

(1) fill the 3 pint cup to its brim,

(2) fill the 5 pint cup to its brim,

(3) empty the 3 pint cup completely into the sink,

(4) empty the 5 pint cup completely into the sink,

(5) pour from the 3 pint cup into the 5 pint cup, and

(6) pour from the 5 pint cup into the 3 pint cup.

For the latter two operations, water is poured until either the source cup is emptied, or the destination cup is filled to the brim, whichever comes first.

The homemaker needs to measure out 1 pint of water. You are to find a way to do this, using exhaustive search. Draw the search tree generated by a **breadth-first** search stopping as soon as the goal state has been reached. Use the notation "(0,0)" to represent the contents of the 3 and 5 pint cups, respectively. The starting state is (0,0), which indicates that both jugs are empty. The goal state is (0,1), indicating that the 3 pint cup is empty, and there is 1 pint of water in the 5 pint cup. In developing the children of each state, apply the transitions in the order given above. Assume that a list is kept of examined states, so that previously examined nodes are not developed a second time.

5. (a) [5 marks] Consider the grammar

```
Z --> A ‡
A -> B
A -> A a B
B -> C
B -> B a C
C -> b
```

It is ambiguous. Show the shortest possible sentence having two different parse trees.

5. (b) [5 marks] What is the language for the grammar in question 5(a) above?

5. (c) [5 marks] Devise an unambiguous grammar having the same language as the grammar in question 5(a) above. Try and keep the number of productions to a minimum.

(d) [5 marks] Given the following grammar,

```
Z  →  E
E  →  T + E
E  →  T
T  →  T * P
T  →  P
P  →  x
P  →  ( E )
```

show the rightmost canonical derivation of the sentence

```
x * (x + x * x)
```

5. (e) [5 marks] Write the productions for a context-free grammar for the following language

$$L(G) = [\ d^{m}\ e\ f^{m}\ \mid\ m>0\ ]$$

6. (a) [12 marks] For the context free grammar whose productions are given below, construct an SLR(1) parser.

Z → S a		*production 0*
S → T c		*production 1*
S → Q b		*production 2*
S → x		*production 3*
Q → R		*production 4*
R → S		*production 5*
T → S		*production 6*

Show your rough work on the back of the previous page, and then fill in the SLR(1) transition table below. Note that the number of states is not totally arbitrary; you must conform to the numbers already assigned. Space has been provided for up to 20 states; if you need more, extend the table. If you detect problems with the grammar, such as ambiguity, show clearly where they occur.

	Q	R	S	T	Z	a	b	c	x
7			8	9					
8									
9									
10									
11									
12									
13									
14									
15									
16									
17									
18									
19									
20									

6. (b) [8 marks] The following finite state machine (FSM) control table has been constructed from a set of 6 production rules, numbered 0 to 5. The start state for the FSM is 6; the halt state is 8.

	a	b	c	d	e	f
6	7			9		
7						8
8	0	0	0	0	0	0
9		10			11	
10				12	13	
11				2		
12			16		14	
13				3		
14		15			11	5
15					13	4
16						1

Complete the productions for this grammar listed below. Each "?" represents a non-terminal or terminal symbol, so the number of elements in the right hand side of each production is given by the number of "?" symbols. [Hint: The symbols "a", "b", etc in the table do not follow the usual convention that they all represent terminals. Some are terminals, but some are non-terminals in the grammar.]

Z → ? ?		*production 0*
a → ? ? ? ?		*production 1*
? → ?		*production 2*
b → ? ?		*production 3*
? → ? ?		*production 4*
? → ?		*production 5*