Professor: Joe Culberson

CMPUT 115(Pascal) Section A2
FINAL EXAM December 19, 2000
CLOSED BOOK. NO Notes or Calculators.
Time 2 Hours.
Answer all questions in space provided.
Do scratch work on page backs.

Extra

Signature_____

Make sure your name and ID is on the top of each internal page

Question 1 *Marks 5* Each part is worth one mark.

1.a What is the running time of the following program fragment? Use Big-O notation.

```
k := n;
while k > 1 do
      k := k - 2;
```

ANSWER:_____

1.b Is heapsort always faster than linear insertion sort? Explain your answer.

ANSWER:_____

REASON:_____

_____

1.c Convert the following infix expression into RPN notation.

$$((A * C) + B + (E/F))$$

ANSWER:_____

1.d Given that $T(n) = n^{25} \log n + 2^n$, write the simplest big-O expression bounding $T(n)$.

ANSWER:_____

1.e Given the declarations

```
type rec = record X,Y: integer; end;
     rptr = ^rec;
var A: array[1..5] of rptr;
```

write a statement to assign 10 to the X field of the 3rd element of the array. (You can assume the records have been created for you).

ANSWER:_____

Question 2 *Marks 3* An email address has the form

```
word@word.word[.word]*
```

where `word` is any sequence of letters from the alphabet `a..z`, the '`@`' and the first '`.`' are required, and the '`[.word]*`' means we can have any number of following words separated by dots. For example, `joe@cs.ualberta.ca` is a valid email address.

Draw a diagram of a finite state machine to recognize email addresses.

Question 3 *Marks 3* What is the output of the following program? Draw diagrams to clearly show how you arrived at your answer, indicating at each line of code what points to what.

```
program point;
uses wincrt;

type ptoi = ^integer;
     ptop = ^ptoi;
var x, y : ptop;

procedure funny(var a: ptop; var b: ptoi);
begin
     a^^ := 5;
     a^ := b;
     new(b);
     b^ := 7;
end;

begin
     new(x);
     new(x^);
     x^^ := 10;
     new(y);
     new(y^);
     y^^ := 15;
     funny(x,y^);
     writeln( x^^,' ', y^^);
end.
```

Question 4 *Marks 7* Consider the following input sequence to heapsort.

1 2 3 4 5 6 7

4.a Draw a sequence of trees showing how phase 1 of heapsort, the bottom up heap construction, works. The first tree should correspond to the input sequence; each following tree should show the partial construction after all siftdowns on the next level of the tree have been performed, with the last of the trees showing the max heap that results.

4.b Show two steps of the second phase of the heapsort algorithm, which put the two largest keys in place. Use trees to illustrate.

Question 5 *Marks 6*

5.a Using the following outputs from two traversals of a binary tree, reconstruct the binary tree. Note it is not a binary search tree.

```
Inorder:   3  5  7  1  2  9  8  4
Postorder: 1  2  3  9  8  7  5  4
```

5.b Show that a preorder traversal and a postorder traversal are not sufficient to allow one to reconstruct a tree.

Question 6 *Marks 6*

6.a Assume you have a hash table with chaining with table size of 10. Draw a diagram showing the result of entering the following sequence of integers, assuming the hash function is $h(i) = i \bmod 10$. You should also assume that list insertions are to the tail of the list.

' 2 5 14 12 22 18 55 34 7

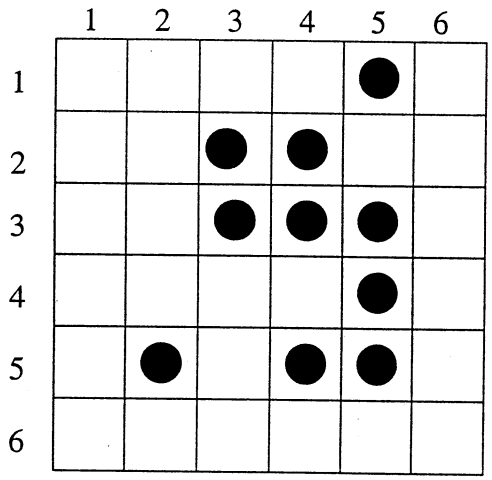
6.b Assume that instead of chaining, you are using the method of double hashing, with the second hash function $h_2(i) = 2i \bmod 10$ if this is non-zero, or otherwise it is 1. Show what happens for the same input sequence as the first part.

6.c What suggestion would you make to improve the double hashing setup?

Question 7 *Marks 9* Consider a square $n \times n$ array in which some cells contain a *dot* and the rest are empty. A *blob* is a maximal set of contiguous dots. Equivalently, a *blob* satisfies the following:

- If a cell contains a dot, then any dot in a cell adjacent (vertically or horizontally) to that dot is also in the same blob.

For example in the following diagram with $n = 6$, there are three (non-empty) blobs; one containing a single dot at $(1, 5)$, another single dot at $(5, 2)$ and a third blob containing the remaining 8 dots. The remaining cells are empty blobs containing no dots.



You are to complete the recursive function `blob(x,y,n:integer):integer` on the next page. This function should return the number of dots in the *blob* starting at cell x, y. n is the size of the array. For example, assuming the above input `blob(1,5,6)` should return 1, and `blob(3,5,6)` should return 8.

To help you along you can assume the following are available:

- a boolean function `isdot(x,y,n:integer):boolean;` which returns true if $1 \leq x, y \leq n$ and cell (x, y) contains a dot. It returns false if the cell does not contain a dot. It is an error to call `isdot` with either x or y less than 1 or greater than n.

- a procedure `markCell(x,y,n:integer);` which marks the cell (x, y), where $1 \leq x, y \leq n$. You may assume that all cells are unmarked initially.

- a boolean function `isMarked(x,y,n:integer):boolean;` which returns true if cell (x, y) is marked, where $1 \leq x, y \leq n$.

The following are hints:

7.a *(1 mark)* For the example, what does `isdot(1,1,6)` return?

ANSWER: _____

7.b *(1 mark)* What should `blob(0,5,6)` return for the example?

ANSWER: _____

7.c *(1 mark)* What should `blob(x,y,n)` return if `isdot(x,y,n)` returns false?

ANSWER: _____

7.d Now complete the following *(6 marks)*.

```
function blob(x,y,n:integer) : integer;
```

Question	Mark	Out Of
1	_____	5
2	_____	3
3	_____	3
4	_____	7
5	_____	6
6	_____	6
7	_____	9
Total	_____	39