Instructor: Ryan Mayward

Sample All Sections
CMPUT 204(algorithms I) Section A1/EA1 A2/EA2
FINAL EXAM December 11, 2000
CLOSED BOOK. NO Notes or Calculators.
Time 2 Hours.

Answer all questions in space provided. Do rough work on backs of pages.

CMPUT 204 (Complete Complete Confinal Pages: 6

extra exam 1

Signature\_\_\_\_

Write your name and ID on the top of EACH internal page

## 39 marks. Justify each answer briefly. Show any necessary calculations.

- 1. (2+3+3=8 marks)
  - (a) Below each function, write the simplest function with the same order of complexity.

$$g \lg n$$

$$n^{2.5} + (n^3)/(\lg^2 n)$$

$$\sum_{j=2}^{n} \lg j$$

$$\sum_{j=n/4}^{n/2} j^2$$

(b) For each of the following functions, give the simplest  $\Theta$  expression (if you can't determine such an expression, give the best O expression you can).

$$f(n) = \begin{cases} 1 & \text{if } n = 1 \\ 7f(\lfloor n/3 \rfloor) + 8n & \text{if } n > 1 \end{cases}$$

$$g(n) = \begin{cases} 1 & \text{if } n = 1\\ 25g(\lfloor n/5 \rfloor) + n^2 & \text{if } n > 1 \end{cases}$$

$$h(n) = \begin{cases} 1 & \text{if } n = 1\\ 7h(\lfloor n/2 \rfloor) + n^3/(\lg n) & \text{if } n > 1 \end{cases}$$

(c) For n keys, give the best case order of complexity for quicksort

selection sort

mergesort

insertion sort

2. (1+1+1+1+3=7 marks)

	a	dj	ac	en	су	matrix					adjacency	list
	1	2	3	4	5	6	7	8	9		•	
1							1		1	1:		
2						1	1			2:		
3							1			3:		
4			1				1			4:		
5							1	1	1	5:		
6		1								6:		
7	1	1	1	1	1				1	7:		
8					1					8:		
9	1				1		1			9:		

- (a) Complete the adjacency list representation of the graph represented above.
- (b) Starting from vertex 1, draw the dfs tree of the graph.
- (c) Starting from vertex 1, draw the bfs tree of the graph.
- (d) A biconnected component of a graph is a maximal biconnected subgraph. Explain what maximal means here.
- (e) Assume that the biconnected component procedure bidfs() has been modified slightly so that outputting each bicomponent is performed by a procedure printC. Draw the procedure call tree for bidfs(1). Each node will be either bidfs(k) for some k or printC. For each printC node, list the edges of the bicomponent in the order in which they are outputted.

3. (1+3+2+1=7 marks)

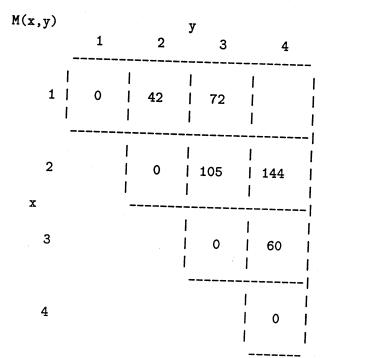
```
1 proc PDSSSP(v) (*Prim/Dijkstra single source shortest paths*)
     D[v] \leftarrow 0
                 (*distance from v: 0*)
     S[v] \leftarrow T
 3
                 (*status:
                               intree*)
 4
     for all x <> v do
 5
      D[x] <- 'infinity'</pre>
 6
       S[x] <- U (*status:
                              unseen*)
     ċ <- v
 8
     loop n-1 times:
      for all nbrs x of t do
 9
10
        if (S[x]=U) or (S[x]=F and _____) then
11
          S[x] <- F (*status: fringe*)</pre>
12
          D[x] <- _____
      t <- any fringe vertex x with smallest D[x]
13
14
      S[t] <- T
                   (*status: intree; best path to t now final*)
```

- (a) Add the missing code at lines 10 and 12.
- (b) For a graph with n vertices and m edges, ignoring the work done in line 13, give a detailed analysis of the running time of this algorithm, assuming that the graph is represented with an adjacency list.
- (c) Assume that each instance of line 13 is implemented by examining all vertices in the graph. Give the total running time of the algorithm.
- (d) Prove that this algorithm is not always correct if edge weights can be negative.

4. (2+2+1+2=7 marks)

 $A_1, \ldots, A_4$  are matrices with dimensions  $2 \times 7$ ,  $7 \times 3$ ,  $3 \times 5$ ,  $5 \times 4$ . M(x,y) is the minimum number of scalar multiplications needed to compute  $A_x \times A_{x+1} \times \ldots \times A_y$ . Showing all details of computation,

- (a) compute M(1,4) [show your work, then put the answer in the table below]
- (b) using parentheses, indicate how to compute  $A_1 \times A_2 \times A_3 \times A_4$  with the minimum number of scalar multiplications. [show your work, then add the parentheses below]



(b) (add parentheses)

A1 A2 A3 A4

- (c) As a function of n, give the time needed to determine M(1, n).
- (d) Q(x,y) is the maximum number of scalar multiplications needed to compute  $A_x \times A_{x+1} \times \ldots \times A_y$ . Give a recurrence relation for Q(x,y).

5. (3 marks)

Using  $\Theta$  notation, give a log cost RAM analysis for the running time of the following.

- 6. (2+5=7 marks) A path in a graph is a sequence of distinct vertices, such that each consecutive pair is adjacent. A path is Hamiltonian if it includes all the vertices of the graph. The problem "given a graph, does it have a Hamiltonian path?" is NP-complete. The k-path problem is: "given a graph and an integer k, does the graph have a path with at least k vertices?". For the k-path problem,
  - (a) either (i) show that it is in NP or (ii) that it is in co-NP.

(b) either (i) show that it is in P or (ii) show that it is NP-complete.