CS 204 Final Wednesday April 17 2002 Prof. Hayward

Show all work. no calculators Time: 3 hours Marks: 39

---

1. (1.5 + 1.5 + 2 marks)

Nessarts's matrix multiplication is similar to Strassen's matrix multiplication: decompose an $n \times n$ matrix into four $n/2 \times n/2$ matrices, perform four recursive $n/2 \times n/2$ matrix multiplications, and recombine the results using ninety-three ordinary $n/2 \times n/2$ matrix additions; in the base 1×1 case, perform one scalar multiplication.

(a) Give a recurrence relation for the number of scalar additions performed in the Nessarts-multiplication of an $n \times n$ matrix. Justify briefly.

(b) Repeat (a) for the total number of scalar operations (additions plus multiplications).

(c) Give the run time to Nessarts-multiply an $n \times n$ matrix. Justify briefly.

2. (1 + 2 + 2 marks)

(a) Let D be a weighted digraph whose arc (directed edge) weights are ab:5, ad:2, bc:3, be:2, ca:8, cd:1, df:1, dg:9, ef:4, eh:1, fb:1, fc:7, fg:8, fh:1, fi:13, fj:3, hi:11, hk:6, jg:4, ji:8, jk:2, ki:5. Draw D below, using the vertices as shown. Draw each arc as an arrow; label each arc with its weight.

```
      b    e    h

a     c    f    i    k

      d    g    j
```

(b) Use Dijkstra's algorithm to find shortest paths in D from vertex a. Show paths by drawing the shortest distance tree on the vertices below; label each vertex with its distance from a.

```
      b    e    h

a     c    f    i    k

      d    g    j
```

(c) Let G be a weighted graph whose (undirected) edge weights are as in (a). Use Prim's algorithm starting from vertex g to find a MST of G. Draw the MST on the vertices below; label vertices in the order they are added to the tree (so g is labelled 1).

```
      b    e    h

a     c    f    i    k

      d    g    j
```

3. (1 + 2 + 3 marks)

```
** Algorithm QQQ: input is an integer array A[1..n]
for j <- n downto 2 do
  **____________________________________________________________
  for k <- 2 to j do
    **__________________________________________________________
    if A[k-1] > A[k] then interchange A[k-1] <-> A[k]
```

(a) State precisely a condition which holds for `A` when the algorithm terminates.

(b) In the pseudocode above, give (i) an outer loop invariant and (ii) an inner loop invariant that could be used to prove (a). (You do not need to prove (a) or the invariants.)

(c) Assuming that `A` initially holds n different numbers, and that each permutation of these numbers is equally likely as the initial input sequence, give an exact expression for the average number of interchanges performed by QQQ. Hint: determine how many interchanges occur for each value of j.

4. (2 + 1.5 + 2.5 marks)

```
0:  5 9
1:  5 7 8 9
2:  6 8
3:  4
4:  3 8
5:  0 1 8
6:  2 8
7:  1
8:  1 2 4 5 6
9:  0 1
```

(a) Beside the above adjacency list representation, starting from vertex 0, draw the graph's (i) BFS tree (ii) DFS tree. Show all non-tree graph edges as dotted lines.

(b) In the bicomponent (biconnected component) algorithm, what is the test to determine whether the parent vertex p of a vertex x cuts off the subtree rooted at x? Explain briefly.

(c) Trace the bicomponent algorithm from class: for each bicomponent, (i) show the edge stack immediately before the edges are popped (ii) list the edges in the order they are popped.

5. (3 + 3 marks)

(a) Suppose that a priority queue is implemented using an unsorted array. Assuming the PQ holds at most k elements, give the time required for (i) delete min (ii) decrease key. Also, if the PQ is used in Dijkstra's SSSP algorithm for a graph with n vertices and $m \geq n$ edges represented with an adjacency list, (iii) give the run time for the algorithm. Justify briefly.

(b) Repeat the above question with "min-heap" instead of "unsorted array".

6. (1 + 2 + 2 marks)

Let G be a connected weighted graph with no two edges having the same weight. Let T_K be an MST of G found by Kruskal's algorithm. Let T_O be a spanning tree of G which contains exactly one edge which is not in T_K; call this edge e_O.

(a) Prove that the graph $T_K + e_O$ contains a cycle C.

(b) Prove or disprove: C contains exactly one edge e_K which is in T_K but not T_O.

(c) Prove or disprove: the weight of T_O is greater than the weight of T_K.

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

A boolean formula f is in k-CNF form if $f = C_1 \wedge C_2 \wedge \ldots \wedge C_t$, where each clause C_j is the OR of exactly k literals; k-CNF-SAT is the problem "given a k-CNF formula, is it satisfiable?"

(a) For the problem 4-CNF-SAT, what are the instances? What is the query?

(b) One way to verify that a formula is satisfiable is to present an assignment of boolean values to boolean variables, and check that each clause is satisfied. Does this verification method imply that 4-CNF-SAT is in NP? Explain briefly.

(c) One way to verify that a formula is not satisfiable is to present every possible assignment of boolean values to boolean variables, and check that the formula is not satisfied. Does this verification method imply that 4-CNF-SAT is in co-NP? Explain briefly.

(d) Prove or disprove: 4-CNF-SAT is in NP-c. You may use any result proved in class or in the textbook.