American University of Beirut

Final *Math 256- Fall 01-02*

Pf. Jana Sukkarieh

February 8th, 02

Name of Student:
Id number:

Let me remind you that this final is not an open notes open books exam like the midterm. Hence, any piece of paper other than your exam booklet and your answer booklet is not allowed. Any blink or whisper results in a zero for the person who tries to cheat and the person who tries to give the answer.

Please read the following guidelines before starting:

- ☞ No questions are allowed during the exam.
- ☞ Write your name and ID on each exam and answer booklet. Hand in the exam booklet with your answer booklet.
- ☞ If you finish your exam before time is up then hand it in to one invigilator and leave. Otherwise ...see next item
- ☞ **IMPORTANT:** When the invigilator says time is up then you stop writing immediately, you stay in your seat. Each student, starting from the last in a 'row', while still seated, passes his/her answer booklet to the one sitting in front of him/her so that all booklets are passed to the first student in that row (which will be collected by an invigilator). You all stay in your seats until all booklets are collected. Anyone who fails to do (for e.g. stand or start talking or anything before) will take a zero. Do not try me!
- ☞ Justify all answers. Show all your work to get partial credit.
- ☞ Use your own judgement (a good one) on how detailed your answer should be(if not specified in the question).
- ☞ **IMPORTANT:** Choose ONE ~~among~~ from exercises 4, 5 and 6. If you do them all, a bonus grade will be given.
- ☞ You have exactly 2 hours and 30 mn
- ☞ **Best of Luck**

1. (5 points) Consider inserting the keys 10, 22, 31, 4, 15, 28, 17, 88, 59 into a hash table of length m=11 using open addressing with the primary hash function h'(k)=k mod m. Illustrate the result of inserting these keys using quadratic probing with $c_1 = 1$ and $c_2 = 3$.

2. (15 points) State then discuss **briefly** the disadvantage(s) of each hashing method (separate chaining, linear probing, quadratic probing, double hashing).

3. (15 points) A recurrent exercise when one starts using graphs is how to color a generic graph (using the least number of colors) to have it in such a way, that no two neighbouring nodes are of the same color (extracted from Nabeel Saad's research paper). Is scheduling final exams at the AUB can be seen as a colouring graph problem ? Note that final exams have to be scheduled so that every student sits for their finals without time conflict or room conflict. If your answer is yes then, assuming that a student is only taking Maths and Computer courses,

   - What exactly does the set of vertices V correspond to?
   - What exactly does the set of edges E correspond to?
   - What exactly do colors correspond to?

4. (15 points) Let e be a maximum-weight edge on some cycle of G=(V,E). Prove that there is a minimum spanning tree of G'=(V,E- {e }) that is also a minimum spanning tree of G.

5. (15 points) Suppose that the graph G=(V,E) is represented as an adjacency matrix. Give a simple implementation of Prim's algorithm for this case that runs in $O(V^2)$.

6. (15 points) Give an example of a directed graph G=(V,E), a source vertex $s \in V$, and a set of tree edges $E_p \subseteq E$ such that for each vertex $v \in V$, the unique path in E_p from s to v is a shortest path in G, yet the set of edges E_p cannot be produced by running BFS on G, no matter how the vertices are ordered in each adjacency list.

7. (25 points) We give pseudocode for 3 different algorithms. Each one takes a graph as input and returns a set of edges T. For each algorithm, you must either prove that T is a minimum spanning tree or prove that T is not a minimum spanning tree

   - First algorithm: Maybe-MST-One(G,w)
     (a) sort the edges into nonincreasing order of edge weights w
     (b) T $\longleftarrow$ E
     (c) for each edge e, taken in non-increasing order by weight
     (d) do if T- {e } is a connected graph
     (e) then T $\longleftarrow$ T- {e}

(f) return T

- Second algorithm: Maybe-MST-Two(G,w)
  (a) T $\longleftarrow \emptyset$
  (b) for each edge e, taken in arbitrary order
  (c) do if T $\cup$ {e } has no cycles
  (d) then T $\longleftarrow$ T $\cup$ {e }
  (e) return T
- Third Algorithm: Maybe-MST-Three(G,w)
  (a) T $\longleftarrow \emptyset$
  (b) for each edge e, taken in arbitrary order
  (c) do T $\longleftarrow$ T $\cup$ {e }
  (d) if T has a cycle c
  (e) then let e' be the maximum-weight edge on c
  (f) T $\longleftarrow$ T- {e'}
  (g) return T

8. (25 points) The goal of the **GRAPH-THEORIST Professor Parone game** is to start with any faculty member at AUB who has worked in a research group, RS_i, and connect these members to Professor Parone in the smallest number of links possible. Two people are linked if they have been members of the same research group at some point (Note then that one person may be in different groups).

   For example, Given that Jana was in the same group RS_1 with Abdul in 1995 and Abdul was in the same group RS_2 with Roger in 1997 and Roger was in the same research group RS_3 with Rodrigo in 1998 and Rodrigo was in the same research group RS_4 as Professor Parone in 1999, we say that Jana is linked to Professor Parone and give this link a Parone-score 4 i.e. minimum number of links that were necessary.

   Given a database of research groups with their members, describe in English how you could compute the Parone-score of all faculty members at AUB. Carefully describe all the major data structures and algorithms you would use in order to solve this problem as efficiently as possible. Write the running time of your algorithm(s) and explain why it is so. Note that each record in the database is a research group name (a string), followed by the number of members in that group, followed by the names of members of that group (a string). You may assume that no two faculty members have the same name and no two research groups have the same names. **Hint:** Model your database as a graph.

9. **Bonus: Trees - Be brief though without jeapordizing essential concepts**

(a) What are red-black trees? What are B-trees? What are binary search trees? Give me some reasons on why/where we use each one of them.

(b) Choose at least one operation that each tree can maintain, write a pseudo-code for the corresponding operation and compare running times with the same operation if maintained by other trees. Argue whether or not so many tree structures are needed.

FINAL ends here ...

Z) **About your own research paper** Remember that what you answer here may change the grade on your research paper dramatically. Make it brief, no jargon i.e. direct to the point and emphasize strengths of your research paper.

- Overview of your paper
- The importance of its topic
- Do you know of any applications in real life or/and computer science?
- What is special about your paper i.e. convince me that you are the one who has done it and understood what you've copied and pasted, if you have!
- What was the benefit (for you that is), if any, out of your research exercise?

Note: Students whose research paper was about red-black trees have to EXPLAIN all operations maintained by red-black trees and compare them with corresponding operations maintained by other kind of trees.