Technische Universität München
Fakultät für Informatik
Lehrstuhl für Effiziente Algorithmen
Prof. Dr. Harald Räcke
Chintan Shah

Winter Semester 2011-12
Final Examination
5th April, 2012

EFFICIENT ALGORITHMS AND DATASTRUCTURES I

Last Name	First Name
....	....

Matrikel No.	Row No., Seat No.	Signature
....	....	....

General Information for the Examination

- Write your name and matrikel no. on all extra supplementaries (Blätter) provided.
- Please keep your identity card readily available.
- Do not use pencils. Do not write in red or green ink.
- You are not allowed to use any device other than your pens and a single sided handwritten A4 sized paper (with your name clearly written on top).
- Verify that you have received 12 printed sides (check page numbers).
- Attempt all questions. You have 120 minutes to answer the questions.

Left Examination Hall from ...... to ...... / from ...... to ......
Submitted Early at ......
Special Notes:

	A1	A2	A3	A4	A5	A6	A7	Σ	Examiner
Maximum	5	3	3	3	8	8	10	40	
1st Correction									
2nd Correction									

Question 1 (5 Marks)

(a) Prove that $\log(n!) = \Theta(n \log n)$.

(b) Give an asymptotic lower bound as well as an asymptotic upper bound on the height of a red-black tree with n nodes.

(c) What is the time taken for consolidating a Fibonacci heap?

(d) Which data structure is more efficient for implementing Prim's MST algorithm: Binary Heap or Fibonacci Heap? Why?

(e) Explain informally, the reasons for primary and secondary clustering.

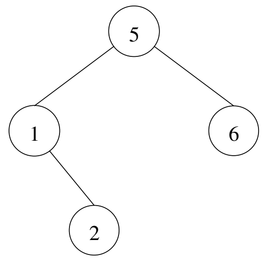
Question 2 (3 Marks)

Solve the following recurrence relation:

$$a_n = a_{n-1} + 2^{n-1} \text{ with } a_0 = 2$$

Question 3 (3 Marks)

Perform the following operations sequentially on the AVL tree shown below so that it remains an AVL tree and show what the tree looks like after each operation (always carry out each operation on the result of the previous operation):



AVL Tree

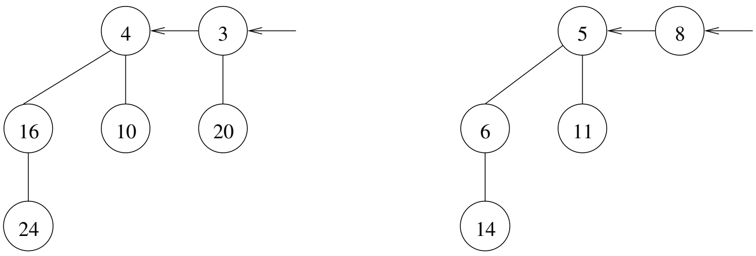
(a) Insert(3)

(b) Insert(4)

(c) Delete(4)

Question 4 (3 Marks)

Perform the following operations sequentially on the Binomial Heaps shown below so that they remain Binomial Heap(s) and show what the Heap looks like after each operation (always carry out each operation on the result of the previous operation):



Heap A　　　　Heap B

(a) A.Insert(1)

(b) A.Merge(B)

(c) A.Delete-Min()

Question 5 (8 Marks)

p families go out to dinner together. To increase their social interaction, they would like to sit at tables so that no two members of the same family are at the same table. Show how to formulate finding a seating arrangement that meets this objective as a maximum flow problem. The ith family has a_i members. There are q tables and the jth table has a seating capacity of b_j.

Question 6 (8 Marks)

An airline has n flight legs that it wishes to service by the fewest possible planes. To do so, it must determine the most efficient way to combine these legs into flight schedules. The starting time for flight i is a_i and the finishing time is b_i. A plane requires r_{ij} hours to travel from the point of destination of flight i to the point of origin of flight j. Show how to solve this problem efficiently by formulating it as a min-cost flow problem.

Question 7 (10 Marks)

A multistack consists of an infinite series of stacks $S_1, S_2, S_3, \ldots$; where S_i can hold upto 2^i elements. We always push and pop elements from the smallest stack S_1. However, before any element can be pushed onto any full stack S_i, we first pop all the elements off S_i and push them onto stack S_{i+1} (one element at a time) to make room. If S_{i+1} is also full, we first recursively move all its members to S_{i+2}. Similarly, before any element can be popped from any empty stack S_i, we first pop 2^i elements from S_{i+1} and push them onto S_i to make elements available for popping. If S_{i+1} is also empty, we first recursively fill it by moving 2^{i+1} elements from S_{i+2}. Moving a single element from one stack to another takes $\Theta(1)$ time.

(a) In the worst case, how long does it take to push one more element onto a multistack containing n elements? (2 marks)

(b) Show that if we never pop anything from the multistack, the amortized cost of a push operation is $O(\log n)$, where n is the maximum number of elements in the multistack during its lifetime. (3 marks)

(c) Show that in any intermixed sequence of push and pop operations, each push or pop operation takes $O(\log n)$ amortized time, where n is the maximum number of elements in the multistack during its lifetime. Use the following potential function:

$$\Phi = \sum_{i=1}^{k} |n_i - 2^{i-1} + 1|$$

where S_k is the largest occupied stack and n_i is the number of elements in stacks smaller than S_i, i.e., the total number of elements in $S_1, \ldots, S_{i-1}$ ($n_1 = 0$). (5 marks)

ROUGH WORK

ROUGH WORK