M.Sc. Qualification exam - 2012

Date: 23.03.12
Time: 3 hours
Answer **4 out of 6 questions**

Important: In the following table, circle the numbers of the four questions you chose to answer.

Question	Grade
1	/25
2	/25
3	/25
4	/25
5	/25
6	/25
Total	

Question 1: Data structures

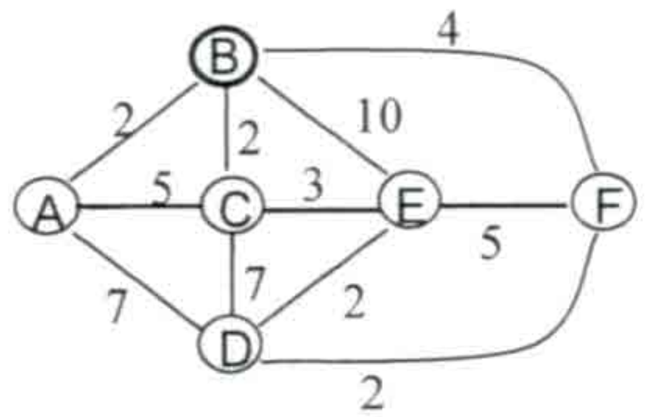
1. Consider a set of n different integers in the range (0,10n);
   a. Suggest an efficient algorithm for constructing a balanced binary search tree with these elements.
   b. What is the worst time complexity of your algorithm?
   c. What is the average time complexity of your algorithm?
2. Consider a set of n different rational numbers in the range (0,1).
   a. Suggest an efficient algorithm for constructing a balanced binary search tree with these elements.
   b. What is the worst time complexity of your algorithm?
   c. What is the best time complexity of your algorithm?
3. Let $T_1(n) = O(f(n))$ and $T_2(n) = \Omega(g(n))$. Give a formal proof or present a counterexample to each of the following claims.
   a. $T_2(n)/T_1(n) = \Omega(g(n)/f(n))$.
   b. $T_1(n) + T_2(n) = O(g(n) + f(n))$

Booklet no_____

I.D no._____

Question 2: Computer Networks

Consider the network shown below:



Fill in the two tables below (in this form, no need to copy to your notebook) with the distance table that would be computed by the distance vector algorithm in node B, once the distance vector algorithm has finished executing.
1. Assume poisoned reverse is not used.
2. Assume poisoned reverse is used.

Answer 1: Table for the case where poisoned reverse is not used.

Table for B	neighbors			
destinations	A	C	E	F
A				
C				
D				
E				
F				

Answer 2: Table for the case where poisoned reverse is used.

Table for B	neighbors			
destinations	A	C	E	F
A				
C				
D				
E				
F				

Question 3: Programming Languages

We are writing an application for managing inventory (מלאי). An inventory consists of items, represented by the following interface:

```
public interface Item {
        // return this item's serial number
        long getSerial();
}
```

The `Inventory` class represents a list of items and their quantities, and has the following interface:

```
public interface Inventory {
        public static class Entry {
                Item item;
                int quantity;
        }

        // Remove all items in the inventory
        public void clear();

        // Add an item to the inventory ("quantity" copies are added)
        public void add(Item item, int quantity);

        // Return an iterator for the items (with quantities) in the
inventory
        public Iterator<Entry> iterator();
}
```

In addition to allowing inventories to be cleared and items added, the inventory API allows the user to iterate over the list of items. This is done by using the `iterator()` function, which returns an iterator object (that implements the standard iterator API, attached below). The order of iteration is the order in which objects were added to the inventory (i.e., the first `next()` call to the iterator returns the first `Entry` added to the inventory, the following `next()` call the second, etc.).

a. [16 pts.] Design and implement in a Java-like language a method to merge inventories with the following signature:

```
public void merge(Inventory joined, Inventory first, Inventory
second);
```

Assume that entries in the `first` and `second` inventories are sorted by the item serial number. (i.e., iterating over the entries in these inventories will return items in order of their serial numbers). After this method is executed, the inventory `joined` should contain all the items in the first and second inventories, also sorted by serial number. If identical items (that have the same serial number) appear in both `first` and `second`, there should only be a single entry in `joined` with that serial number, but the quantity should be the sum of the two original quantities.

No points will be deducted for syntax errors. Instead, what we are looking for in this question is elegant pseudo code.

b. [9 pts.] Fork and Spoon are both concrete classes that implement Item (but neither is a subclass of the other). Which of the following code fragments will cause errors? (can be more than one). Explain (in one line) what will cause the error.

i. Item[] items = new Item[8]; items[10] = new Fork();

ii. Spoon[] spoons = new Spoon[8]; spoons[4] = new Fork();

iii. Item[] items = new Item[8]; items[3] = new Spoon();

iv. Spoon[] spoons = new Item[8]; spoons[2] = new Spoon();

v. Item[] items = new Spoon[8]; items[1] = new Fork();

Interface Iterator<E>

Method Summary	
boolean	hasNext() Returns `true` if the iteration has more elements.
E	next() Returns the next element in the iteration.

Method Detail

hasNext

boolean **hasNext**()

Returns `true` if the iteration has more elements. (In other words, returns `true` if `next` would return an element rather than throwing an exception.)

Returns:

`true` if the iterator has more elements.

---

next

E **next**()

Returns the next element in the iteration.

Returns:

the next element in the iteration.

Throws:

NoSuchElementException - iteration has no more elements.

Question 4: Algorithms

1. (10 pts.) Let G=(V,E) be a simple connected undirected graph. BFS is performed on the graph |V| times, each time a different vertex is the root.
   It is known that for some edge e∈E, e=(u,v), it holds that e belongs to all the |V| spanning trees produces by the different runs.

   Select the correct claim and prove it.
   1. The edge e does not belong to any cycle.
   2. The edge e might belong to a cycle. However, if e belongs to an odd-length cycle, then it must belong also to an even-length cycle.
   3. The edge e might belong to a cycle. However, If e belongs to an even-length cycle, then it must belong also to an odd-length cycle.
   4. None of the above is correct.

   Remark: The length of a cycle is the number of edges it includes.

2. (15 pts.) Given is a flow network in which all the capacities are integers. An algorithm for finding a maximum flow is run on the network and produces a max-flow of value F_1. The capacity of a specific edge e∈E is reduced by x. The max-flow in the resulting network is F_2.
   Prove or disprove each of the following claims (separately).

   a. If $F_1=F_2$ then in the initial flow the edge e is not saturated.
   b. If before the capacity reduction the edge e is on some min-cut of the network, then $F_2=F_1-x$.

Question 5: Operating Systems

Modern operating systems support the process of Hibernation – in this process, the current state of the computer is saved, along with all opened programs and running background processes, and the computer is being completely turned off (no power supply). When the computer is turned on later the system returns to the exact state as it was before it was turned off. The operation is done by saving the memory space of all processes to the disk during shut down and reloading it into memory during system startup.

For each of the following claims – answer whether it is true or false and explain:

A. When turning the computer off, only the pages of which the "modified" bit is on should be saved to disk.

B. When turning the computer on, only the pages of which the "valid" bit was on during shut down should be loaded into main memory.

C. The length of the turn-off operation grows with the size of the physical memory (for all processes together).

D. The length of the turn-off operation grows with the size of the virtual space (for all processes together).

Question 6: Computability and Complexity

Note: you need to prove your answers.

a. Consider the language

$$L_1 = \{<M> |\ M \text{ is a TM and } \forall w \in \Sigma^* \text{ if } M \text{ accepts } w \text{ then } M \text{ does not accept } Rev(w)\}$$

where $\text{Rev}(w_1 w_2 \ldots w_n) = w_n w_{n-1} \ldots w_1$. Is $L_1 \in R$? Define $\overline{L_1}$. Is $\overline{L_1} \in RE$?

b. A graph G=(V,E) is said to be k-colorable if its vertices can be colored using k colors so that no two neighboring vertices in G have the same color. (More formally, if there exists a function $\pi : V \to \{1, 2, \ldots, k\}$ so that for all edges $(i, j) \in E$ it holds that $\pi(i) \neq \pi(j)$). Let

$$COL = \{<G,k> | G \text{ is a simple graph, and } G \text{ is } k\text{-colorable}\}$$
$$EVEN\text{-}COL = \{<G,k> | G \text{ is a simple graph, } k \text{ is even, and } G \text{ is } k\text{-colorable}\}.$$

In this question you may assume COL is NP-hard. Show that if EVEN-COL is in P then P=NP.

GOOD LUCK