FACULTY OF
SCIENCE & TECHNOLOGY

# COMPUTER SCIENCE AND SOFTWARE ENGINEERING

## 2013-2014

| | | | |
|---|---|---|---|
| **Code:** | ECSE502 | **Level:** 5 | **Semester:** 2 |
| **Title:** | Algorithms and Data Structures | | |
| **Date:** | 12 May 2014 | | |
| **Time:** | 10:00 | | |
| **Duration:** | 2 Hours | | |
| **Module Leader:** | Epaminondas Kapetanios | | |

## INSTRUCTIONS TO CANDIDATES

This paper contains seven questions in section A and seven problem statements in section B. You need to answer five out of seven questions in section A and address ALL problems in section B.

Information sheets and additional stationery supplied:

N/A

## SECTION A (20 marks):
## Answer any five (5) of the following seven (7) questions

1. What is the time complexity, in terms of the Big-O notation, when one inserts an item into an unordered array? Justify your answer.

[4 marks]

2. What is the time complexity of the *search* operation, in terms of the Big-O notation, in an unordered array which allows duplicates? Justify your answer.

[4 marks]

3. What is the maximum number of comparisons (worst case scenario) in order to complete a binary search in an array of 256 elements, if the search key cannot be found on the array? Justify your answer.

[4 marks]

4. Let us assume that the underpinning data structure for the implementation of queues is an array. Explain how will you guarantee that an empty queue (array) is not taken mistakenly for a full one.

[4 marks]

5. Which link(s) in a doubly linked list grants access to the whole list? Justify your answer.

[4 marks]

6. Given a search key, what kind of comparison applies when we are interested in navigating through the binary tree in order to locate the search key on a particular node?

[4 marks]

7. How can we maintain or restore balance in an unbalanced tree?

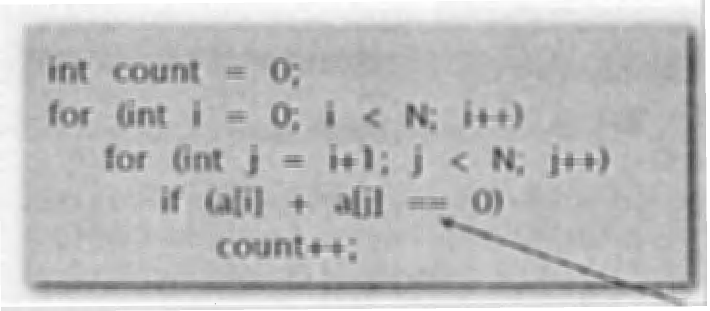
[4 marks]

## SECTION B (80 marks):
## Provide solutions for all the following problems

### PROBLEM 1:

The following piece of code implements in Java the two-sum algorithm checks, i.e., checking whether the sum of any pairs of numbers on an array of N integers equals zero. Based on the lines of code, estimate and justify its complexity by considering how many times all major operations (e.g., variable declarations, value assignments, comparisons, increments, array accesses) will be executed.

[12 marks]



```
int count = 0;
for (int i = 0; i < N; i++)
    for (int j = i+1; j < N; j++)
        if (a[i] + a[j] == 0)
            count++;
```

Statement checking whether the sum equals to zero

### PROBLEM 2:

Let us assume you run an algorithm by doubling the input size N and having recorded the following times spent on every run of the algorithm. Estimate the time complexity in terms of the well-established orders of growth and justify your answer.

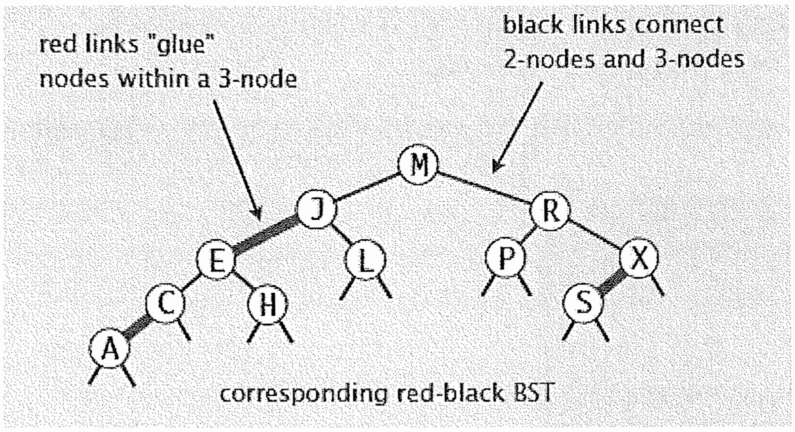
[10 marks]

```
       N   seconds   ratio of change  logarithmic ratio
-----------------------------------------------------
     512      0.12
    1024      0.24       2                   1
    2048      0.48       2                   1
    4096      1          2                   1
```

## PROBLEM 3:

Study the red-black tree below and transform it into its corresponding 2-3 balanced tree by taking into consideration the role of the red and the black links.

[8 marks]


red links "glue" nodes within a 3-node
black links connect 2-nodes and 3-nodes
M
J
R
E
L
P
X
C
H
S
A
corresponding red-black BST

## PROBLEM 4:

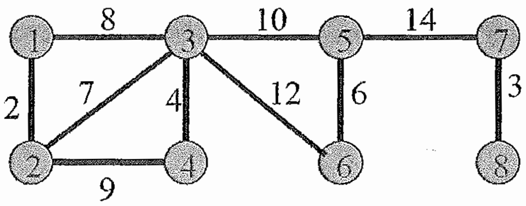
Let us assume you are given four friends, {Sonya, Anna, Matilda, Dora}, and the relationships {*Sonya likes Anna*}, {*Anna likes Sonya, Matilda, Dora*}, {*Matilda likes Anna*}, {*Dora likes Anna*}. Suggest a data structure and its two alternative implementations capable of representing the knowledge about the four friends above.

[12 Marks]

## PROBLEM 5:

Given the following graph, describe and apply Kruskal's method in order to derive the minimum spanning tree (MST) from this graph? What will be the MST derived from the graph?

[16 Marks]



## PROBLEM 6:

Let us assume that you are given an unsorted list [8, 3, 13, 6, 2, 14, 5, 9, 10, 1, 7, 12, 4], which needs to be sorted into the list [1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 12, 13, 14] by applying the merge-sort algorithm. Describe the algorithm in pseudo-code, Java or C++, which implements the merge-sort algorithm **iteratively**, NOT recursively.

[12 Marks]

## PROBLEM 7:

Let us assume that the following notation stands for times spent in minutes and connections among cities, represented by the letters A, B, C, D, E.

A->B: 50, A->D: 80, B->C: 60, B->D: 90, C->E: 40, D->C: 20, D->E: 70, E->B: 50

For instance, you can go from city A to city B in 50 minutes. Given this knowledge, we need to figure out what is the travel path from city A to city E with the least time spent. Subsequently,

a) name the algorithm and data structure you should apply to resolve this problem,
b) specify the path to follow and the least time spent, in order to move from city A to city E.

[10 Marks]

END OF EXAM PAPER