Course "Data Structures and Algorithms" – PART1

Prof. Roberto Sebastiani

February 17^{th}, 2011

Test code:	26
Name:	**Sonia**
Surname:	**Frei**
Matriculation #:	10110350

Instructions: for each question, insert in the following table the right answer (A,B,C,D) or leave it blank.
The answers must be clear, understandable and unambiguous. Only the content of the table will be considered for the evaluation.
Score: The score of the text part is computed as follows. For each multiple-answer question Q.i, the score $score(Q.i)$ is:

- 3 points if the answer is correct
- -1 points if the answer is wrong
- 0 points if unanswered

Then the score, expressed in 30's, is computed as:

$$score = round(0.5 \cdot \sum_{i=1}^{22} score(Q.i)).$$

The test is passed if $score \geq 18$. If $score \in [15, 17]$, then the student can ask for having the second part corrected, at his/her own risk.

Examples:

- 22 correct answers: 33 ($30L$) $\Longrightarrow$ passed
- 13 correct + 3 wrong answers: 18 $\Longrightarrow$ passed
- 14 correct + 8 wrong answers: 17 $\Longrightarrow$ failed (can ask)
- 12 correct + 9 wrong answers + 1 unanswered: 14 $\Longrightarrow$ failed

Q.#:	ANSWER:
01	[Solution: B]
02	[Solution: A]
03	[Solution: D]
04	[Solution: A]
05	[Solution: D]
06	[Solution: A]
07	[Solution: B]
08	[Solution: B]
09	[Solution: A]
10	[Solution: C]
11	[Solution: B]
12	[Solution: C]
13	[Solution: A]
14	[Solution: D]
15	[Solution: D]
16	[Solution: A]
17	[Solution: A]
18	[Solution: D]
19	[Solution: B]
20	[Solution: A]
21	[Solution: A]
22	[Solution: C]

This page is willingly left empty.

Notation and conventions: *hereafter, if not otherwise specified, the variable "N" denotes the number of elements contained in a data structure (list, stack, queue,etc.) or the size of an array. Also, for arrays and binary search trees we assume the increasing order, if not otherwise specified. For Heaps and Priority Queues, we assume the same conventions used in the classes.*

Q.01 Consider the following pseudo-code of the function `sum`:

```
INPUT:  n, m two natural numbers larger or equal to 0.
OUTPUT: n+m, the sum of n and m.

sum(n,m)
if (n == 0 )
  then return m
else
  return sum(n-1,m+1)
```

Only one of the following facts is true. Say which one.

A. The function `sum` computes correctly the sum of `n` and `m`, and it requires a constant amount of recursive calls; [Solution: NO]

B. The function `sum` computes correctly the sum of `n` and `m`, and it requires an amount of recursive calls which is linear on `n` [Solution: YES]

C. The function `sum` computes correctly the sum of `n` and `m`, and it requires an amount of recursive calls which is linear on `m` [Solution: NO]

D. The function `sum` does not compute correctly the sum of `n` and `m`; [Solution: NO]

[Solution: B]

Q.02 A given sorting algorithm is applied to the array: `[ 44 55 12 42 94 18 16 67 ]`. The following is the evolution of the array, each step representing one external loop of the algorithm:

```
[ 44 55 12 42 94 18 16 67 ] // before the 1st external loop
[ 12 55 44 42 94 18 16 67 ] // after the 1st external loop
[ 12 16 44 42 94 18 55 67 ] // after the 2nd external loop
[ 12 16 18 42 94 44 55 67 ] // after the 3rd external loop
[ 12 16 18 42 94 44 55 67 ] // after the 4th external loop
[ 12 16 18 42 44 94 55 67 ] // after the 5th external loop
[ 12 16 18 42 44 55 94 67 ] // after the 6th external loop
[ 12 16 18 42 44 55 67 94 ] // after the 7th external loop
```

Only one of the following facts is true. Say which one.

A. The sorting algorithm is SelectionSort [Solution: YES]

B. The sorting algorithm is MergeSort [Solution: NO]

C. The sorting algorithm is QuickSort [Solution: NO]

D. The sorting algorithm is InsertionSort [Solution: NO]

[Solution: A]

Q.03 Which of the following functions $f(N)$ best represents the number of steps performed by Linear Search over a sorted input array of size N?

A. $f(N) = O(log(N))$ [Solution: NO]

B. $f(N) = O(N^2)$ [Solution: NO]

C. $f(N) = O(N \cdot log(N))$ [Solution: NO]

D. $f(N) = O(N)$ [Solution: YES]

[Solution: D]

Q.04 Consider the following piece of pseudo-code:

```
INPUT: A generic array A of range 1...N of integer values
OUTPUT: the maximum value of all elements of A;
  max = A[N]; i=N-1;
  WHILE (i >= 0)  {
     if (a[i]>max)
        max = A[i] ;
     i=i-1;
  }
  return max;
```

Only one of the following statements is a loop-invariant for the "for" loop, allowing to prove the correctness of the function sum. Say which one. (Note: we conventionally assume w.l.o.g. that "after the 0-th iteration" here means "before for loop starts".)

A. when evaluating the "WHILE" condition, `max` contains the maximum element of the sub-array $A[i+1, ..., N]$ [Solution: YES]

B. when evaluating the "WHILE" condition, `max` contains the maximum element of the sub-array $A[1, .., i+1]$ [Solution: NO]

C. when evaluating the "WHILE" condition, `max` contains the maximum element of the sub-array $A[i, ..., N]$ [Solution: NO]

D. when evaluating the "WHILE" condition, `max` contains the maximum element of the sub-array $A[1, .., i]$ [Solution: NO]

[Solution: A]

Q.05 The function $f(N)$ describing the maximum recursion depth of MergeSort wrt. the size N of the array is:

A. $\Theta(N \cdot log(N))$ [Solution: NO]

B. $\Theta(N)$ [Solution: NO]

C. $\Theta(N^2)$ [Solution: NO]

D. $\Theta(log(N))$ [Solution: YES]

[Solution: D]

Q.06 Consider the recurrence $T(n) := 16T(n/2) + 128n^3\sqrt{n}$. According to Master Theorem, only one of the following facts is true. Say which one.

A. $T(n) = \Theta(n^4)$ [Solution: YES]

B. Master Theorem does not allow to conclude anything about $T(n)$. [Solution: NO]

C. $T(n) = \Theta(n^4 log(n))$ [Solution: NO]

D. $T(n) = \Theta(n^3\sqrt{n})$ [Solution: NO]

[Solution: A]

Q.07 Only one of the following facts about Binary Search is *false*. Say which one.

A. Can be implemented iteratively without using an explicit stack. [Solution: NO]

B. Requires the size of the input array to be a power of two. [Solution: YES]

C. On sorted arrays, it is typically much more efficient than linear search. [Solution: NO]

D. If applied to an un-sorted array, it may return a wrong answer. [Solution: NO]

[Solution: B]

Q.08 Only one of the following facts about HeapSort is *false*. Say which one.

A. It based on the same macro-schema as SelectionSort. [Solution: NO]

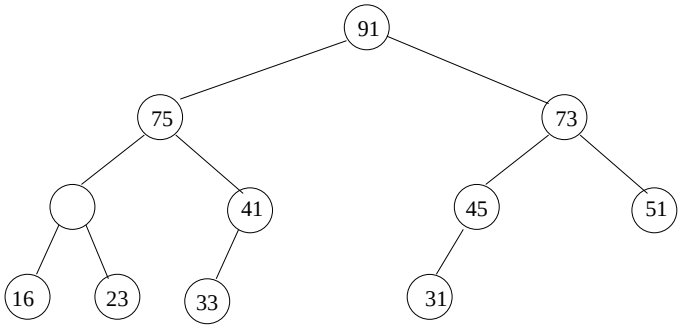
B. It has the same complexity as SelectionSort. [Solution: YES]

C. It takes $O(n \cdot log(n))$ [Solution: NO]

D. On everage, it is less efficient than QuickSort [Solution: NO]

[Solution: B]

Q.09 Consider the following binary tree, with a blank node:



Only one of the following facts is true. Say which one.

A. The tree is not a heap, no matter the value of the blank node. [Solution: YES]

B. If the value of the blank node is 60, then the tree is a heap. [Solution: NO]

C. If the value of the blank node is 22, then the tree is a heap. [Solution: NO]

D. If the value of the blank node is 76, then the tree is a heap. [Solution: NO]

[Solution: A]

Q.10 Only one of the following facts about QuickSort, in the standard basic version we have seen in the class, [1] is *false*. Say which one.

A. It is a recursive algorithm. [Solution: NO]

B. If the input array is already sorted, it takes $O(N^2)$ time to execute. [Solution: NO]

C. It always takes $\Theta(N \cdot log(N))$ time to complete. [Solution: YES]

D. It is the fastest sorting algorithm on average. [Solution: NO]

[Solution: C]

Q.11 Consider the standard implementation of a Stack as an linked-list. Only one of the following facts is true. Say which one.

A. Both the insertion and the extraction of one element require $O(N)$ steps. [Solution: NO]

B. Both the insertion and the extraction of one element require $O(1)$ steps. [Solution: YES]

C. The insertion of an element requires $O(1)$ steps, whilst the extraction of an element requires $O(N)$ steps. [Solution: NO]

D. The insertion of an element requires $O(N)$ steps, whilst the extraction of an element requires $O(1)$ steps. [Solution: NO]

[Solution: B]

[1] that is, s.t. it selects the rightmost element of the sub array as pivot, no randomization.

Q.12 Consider the following piece of JAVA code:

```
for (i=0;i<A.length();i++)
  Q.enqueue(A[i]);
while (! Q.IsEmpty())
  S.Push(Q.dequeue());
j=0;
while (! S.IsEmpty()) {
  B[j]=S.pop();
  j++;
}
```

where `S` is a stack, `Q` is a queue, `A, B` are two arrays of the same length, and before the for loop `A` contains:

`[ 28 13 81 44 33 7 ]`.

Then after the execution of the while loop `B` contains:

A. `[ 28 13 81 44 33 7 ]`. [Solution: NO]

B. `[ 7 13 28 33 44 81 ]`. [Solution: NO]

C. `[ 7 33 44 81 13 28 ]`. [Solution: YES]

D. `[ 81 44 33 28 13 7 ]`. [Solution: NO]

[Solution: C]

Q.13 Consider the standard operation `Heapify(A,i)` on a heap `A` of size `N`, `i` $\in$ `[1..N]`. Only one of the following facts is true. Say which one.

A. In order to correctly apply `Heapify(A,i)`, the binary trees rooted on left(i) and right(i) must be heaps [Solution: YES]

B. While executing `Heapify(A,i)`, both binary trees rooted on left(i) and right(i) are modified. [Solution: NO]

C. `Heapify(A,i)` takes $\Theta(N)$ to execute. [Solution: NO]

D. In order to correctly apply `Heapify(A,i)`, the value of the element of index i must be greater than those of left(i) and right(i). [Solution: NO]

[Solution: A]

Q.14 Consider the following piece of JAVA code:

```
for (i=0;i<A.length();i++)
  Q.insert(A[i]);
while (! Q.IsEmpty())
  S.Push(Q.extractMax())
i=0;
while (! S.IsEmpty()) {
  B[i]=S.Pop();
  i++;
}
```

where `S` is a stack and `Q` is a priority queue, `A`, `B` are arrays, and before the for loop `A` contains:

`[ 28 13 81 44 33 7 ]`.

Which of the following content is printed?

A. `[ 81 44 33 28 13 7 ]`. [Solution: NO]

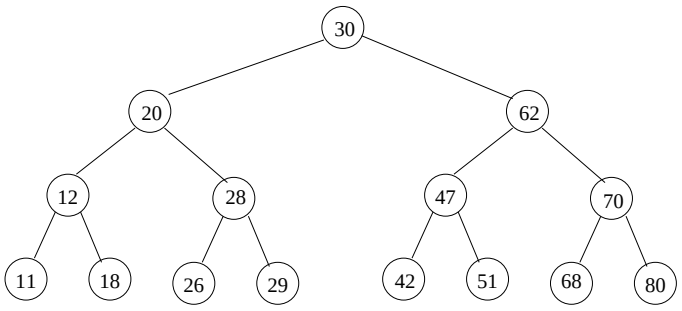
B. `[ 7 33 44 81 13 28 ]`. [Solution: NO]

C. `[ 28 13 81 44 33 7 ]`. [Solution: NO]

D. `[ 7 13 28 33 44 81 ]`. [Solution: YES]

[Solution: D]

Q.15 A binary search tree of integer values is initially empty. Then, after inserting in order a given sequence of integers, the tree looks like the following.



Only one of the following input sequences of integer values may have produced the tree above. Say which one.

A. 11 12 18 20 26 28 29 30 42 47 51 62 68 70 80 [Solution: NO]

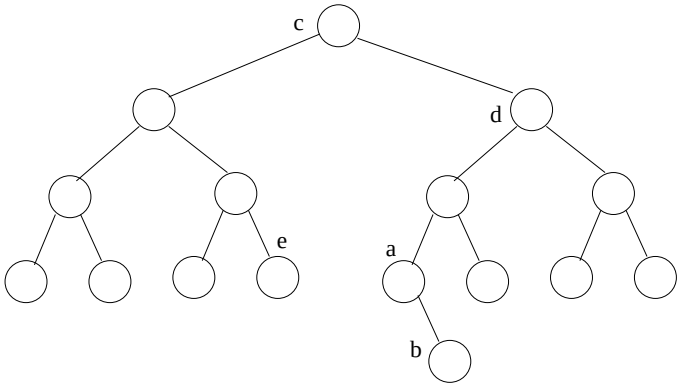
B. 11 18 26 29 42 51 68 80 12 28 47 70 20 62 30 [Solution: NO]

C. 80 70 68 62 51 47 42 30 29 28 26 20 18 12 11 [Solution: NO]

D. 30 62 20 70 12 28 47 18 11 26 29 51 42 68 80 [Solution: YES]

[Solution: D]

Q.16 Consider the following binary search tree, representing a sequence of integer values in increasing order, s.t. the values have been hidden. (Note: the labels "a"..."e" are only conventional names we give to indicate them.)



Which of the following nodes is the predecessor of node "a"?

A. Node "c". [Solution: YES]

B. Node "b". [Solution: NO]

C. Node "e". [Solution: NO]

D. Node "d". [Solution: NO]

[Solution: A]

Q.17 Only one of the following facts about binary search trees is true. Say which one.

A. The search in a binary search tree is at most linear wrt. the height of the tree. [Solution: YES]

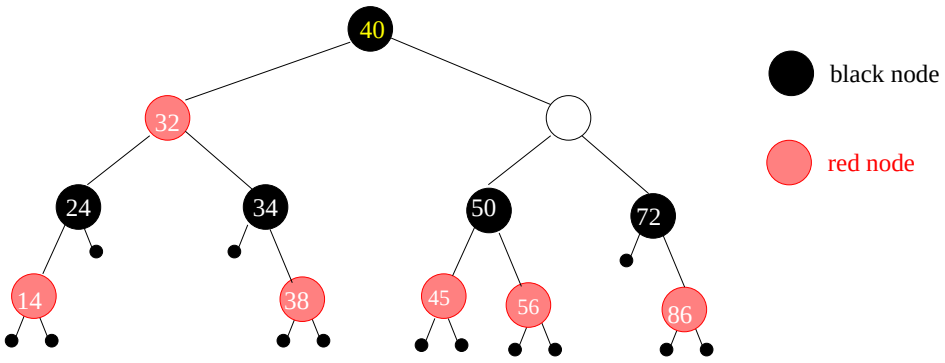
B. The search for an element always requires $\Theta(N)$ steps, N being the number of elements contained in the tree. [Solution: NO]

C. If a binary search tree is created from an empty one by inserting one-by-one a sequence of elements in decreasing order, that the resulting tree is balanced. [Solution: NO]

D. The search for an element always requires $O(log(N))$ steps, N being the number of elements contained in the tree. [Solution: NO]

[Solution: A]

Q.18 Consider the following binary tree with a blank node:



Only one of the following facts is true. Say which one.

A. If the blank node is black and its value is 65, then the tree is a RB-tree. [Solution: NO]

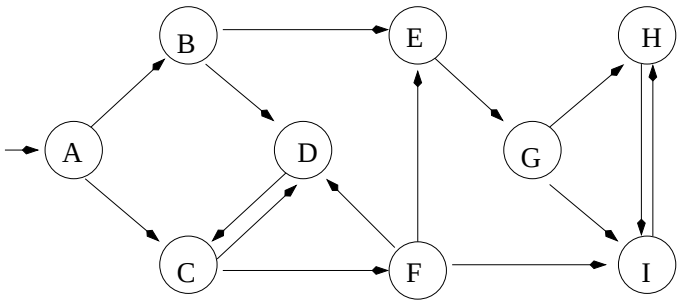
B. If the blank node is red and its value is 54, then the tree is a RB-tree. [Solution: NO]

C. The tree cannot be an RB-tree, no matter the color and value of the blank node. [Solution: NO]

D. If the blank node is red and its value is 62, then the tree is a RB-tree. [Solution: YES]

[Solution: D]

Q.19 Given the following directed graph:



Only one of the following sequences of arcs represents a visit of the graph in depthth-first search (DFS) starting from the node A. Say which one. (Notation: e.g., "AB" means the arc "$A \longrightarrow B$".)

A. AB AC BE BD CF EG FI GH [Solution: NO]

B. AB BE EG GH HI IH GI BD DC CD CF FD FE FI AC [Solution: YES]

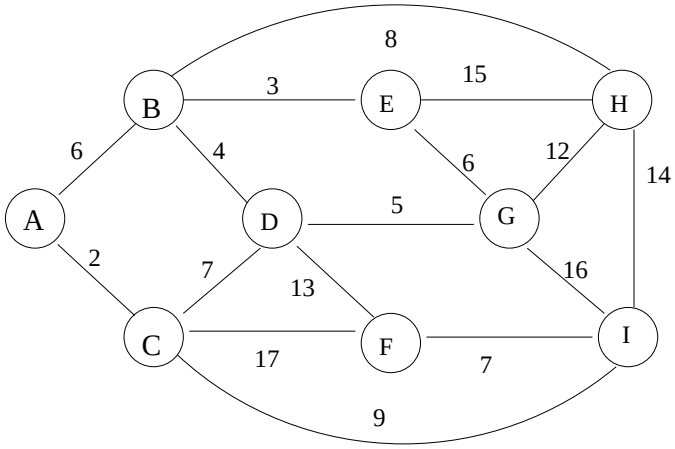
C. AB AC BE BD CD CF EG DC FD FE FI GH GI IH HI [Solution: NO]

D. AB BE EG GH HI BD DC CF [Solution: NO]

[Solution: B]

Q.20

Consider the following undirected weighted graph:



Only one of the following sets of arcs represents a minimum spanning tree. Say which one.

A. { AC AB BE BD DG BH CI IF }
[Solution: YES]

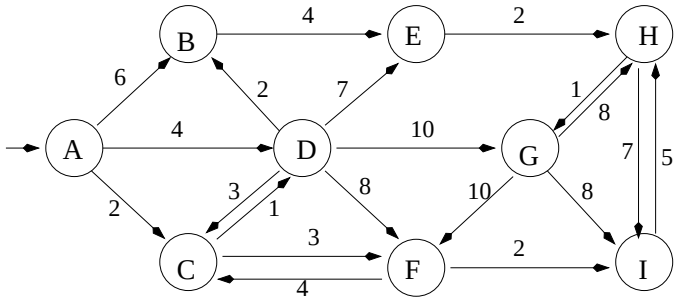
B. { AC EG BD BE DG BH CI CD } [Solution: NO]

C. { AB AC BE BD CF EH EG FI } [Solution: NO]

D. { AC BE BD DG AB EG FI CD } [Solution: NO]

[Solution: A]

Q.21 Consider the following directed weighted graph:



Only one of the following sets of arcs represents a shortest path tree starting from node A. Say which one.

A. AC CD DB CF FI BE EH HG [Solution: YES]

B. AC CD DB CF DE FI EH HG [Solution: NO]

C. AB AD AC DB DF DE DG GI HI [Solution: NO]

D. AC CD AB CF FI BE EH HG [Solution: NO]

[Solution: A]

Q.22

At the end of the execution of the Belman-Ford algorithm on a weighted directed graph $G \stackrel{\text{def}}{=} (V, E)$, we have that, for one edge $v_i \longmapsto v_j$, the following fact holds: $d(v_j) > d(v_i) + w(v_i, v_j)$. What can we conclude from this fact?

A. the shortest path to v_j passes through $v_i \longmapsto v_j$; [Solution: NO]

B. the shortest path to v_j does not pass through $v_i \longmapsto v_j$; [Solution: NO]

C. there is a negative cycle in the graph G; [Solution: YES]

D. there is no negative cycle in the graph G; [Solution: NO]

[Solution: C]