

Computer Science Foundation Exam

March 9, 2001

Section I A

solutions

(1, 20%) a) Show the array **x** after the procedure has completed execution (2pts each)?

Array x	15	12	5	7	11	17
position	1	2	3	4	5	6

b) What value will the variables **a** and **b** contain when the algorithm is finished (4pts each)?

a	9	b	8
----------	----------	----------	----------

(2, 14%) a) $7^2 \cdot 4 - 3^A * 5 + 8^B \cdot 4 / * 6^C + - = 3$

3
-2
7

A

8
-1
7

B

6
-2
7

C

b) output from print statements:

2	7	3	4
first output	second output	third output	fourth output

(3, 20%)

a) For an $O(n!)$ algorithm, one data set with $n = 4$ takes **72** seconds.

How long will it take for a data set with $n = 5$?

360 seconds

b) For an $O(2^n)$ algorithm, one data set with $n = 7$ takes **96** seconds.

If you used a different-sized data set and it took

12 seconds, how large must that data set be?

n = 4

c) For an $O(n^3)$ algorithm, a friend tells you that it took **192** seconds to run on her data set. You run the same program, on the same machine, and your data set with $n = 6$ takes **648** seconds.

What size was her data set?

n = 4

d) What is the Order of this pseudocode segment?

$O(n^2)$

e) What will be the value of **x** when the **for** loops end?

$2n^3 + 5n^2 + 2n$

(4, 10%) Write a **recursive** function, called **prob4**, that will correctly print the index positions of all the occurrences of a specified character **c**, within the first **m** locations of an array **X**. You may assume that **X** is a global array which includes locations that range from **1** to **n** and is already populated with characters. Assume that $m \leq n$. The initial call is **prob4(c, m)**. You may use pseudocode, C, Java or Pascal syntax but points will be deducted if your meaning is not clear.

Two of the many ways that this can be done:

```
procedure prob4(c : char, m : integer) {  
    if (m > 0) {  
        if (X[m] == c)  
            print("Found ", c, " at ", m)  
        prob4(c, m-1)  
    }  
}
```

OR

```
procedure prob4(c : char, m : integer) {  
    if (m > 0) {  
        prob4(c, m-1)  
        if (X[m] == c)  
            print("Found ", c, " at ", m)  
    }  
}
```

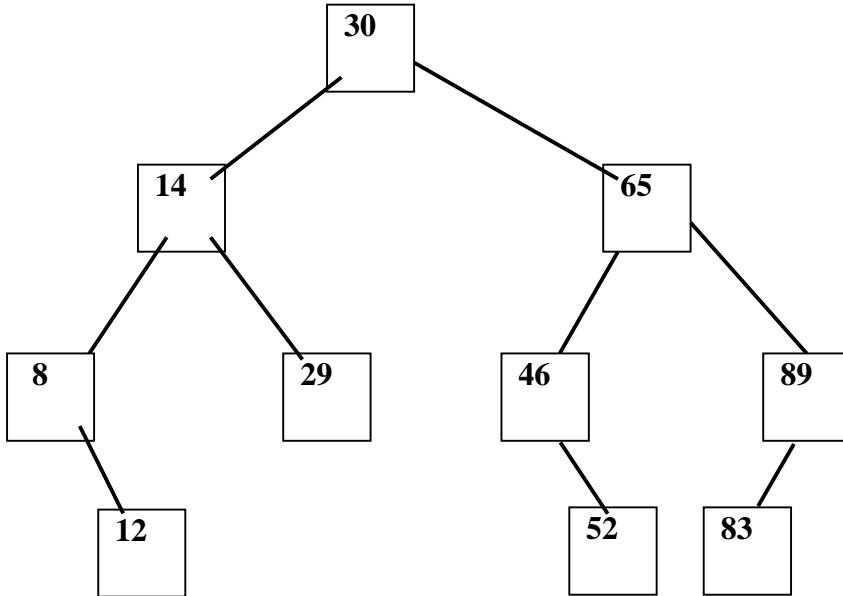
(5, 18%) Find the closed form or exact value for the following:
(*n* is an arbitrary positive integer):

a) $\sum_{i=0}^{47} (3i+1) = \mathbf{3432}$

b) $\sum_{i=1}^{2k-2} (2i+5) = \mathbf{4k^2 + 4k - 8}$

c) $\sum_{i=25}^{75} (4i-3) = \mathbf{10047}$

(6, 18%) Given the following Binary Tree, answer the questions below :



a) Is this a valid Binary Search Tree? (circle one)

Yes

b) List the nodes of this tree in the order that they are visited in a **preorder** traversal:

30	14	8	12	29	65	46	52	89	83	
first node visited										last node visited

c) Perform the following procedure on the tree above, listing the output in the spaces below and leaving any unused spaces blank. Assume that the procedure is initially called with **P6(root, 30)**

12 29 14 52 89