Smart Coding & Interview Series
Top20-Basic Test Series
(Online Test - I)

General Instructions **Duration: 75 minutes**

1. Please go through all the instructions carefully.
2. The test can be taken in any language of your choice - 'C', 'C++', 'Java' or 'Python
3. Note that a more optimal solution in terms of time-/space-complexity will be judged superior.
4. Do no use static or global variables in any code you are writing.
5. The purpose of these test discussion is to present the exact reality of online test and interview questions asked in top-notch companies.
6. The thought processes discussed in the top20-basic, top20-advanced and top20-pro programs must give enough confidence and depth to solve/crack such tests and interviews with ease. Transformation in thought process is required instead of byhearting FAQs. FAQ solution remembering will not take you far in your career.
7. The standards of tests will be much higher for top20-advanced and top20-pro programs. They will be discussed as part of the corresponding programs.
8. This test consists of 4 sections: Data structure Applications, Time & Space Complexity, Bug Fixing and Subjective question coding.
9. These questions are actual questions asked in tests & interviews of top-notch companies like Google, Microsoft, Amazon, Facebook, Twitter, Wallmart Labs, etc., across the globe.
10. This is the scientific and standard format most software product companies follow to pick right candidates for job.
11. This format focuses only on test and interview questions related to data structure analysis, data structure design, algorithm design, algorithm analysis, coding and bug fixing.
12. The formats related to language skill, design & architecture, artificial intelligence are discussed as part of the relevant series and courses presented at:

    www.algorithmicaonline.com

algorithmica

Smart Coding & Interview Series
Top20-Basic Test Series
(Online Test - I)

Section A: (4 x 2 mins) = 8 mins
This section contains questions relating to application of data structures.

1. Which of the following is least suitable for implementing a set of integers, in terms of time complexity for the operations performed in it, given that main operation in the set is finding whether an element is present in it or not and assuming the maximum value that can be stored in the set is fixed?
   a. Binary Tree
   b. Heap
   c. Bit Vector
   d. Hash Table

2. Which of the following data structure can be used to represent stack, queue, binary tree, hash table, heap & graph?
   a. Array
   b. Singly Linked List
   c. Double Linked List
   d. None of the above

3. You require a data structure that supports the following operations:
   - Insert(x) : inserts element 'x' if it does not exist.
   - Delete(x) : deletes the element 'x' if it exists
   - FindNext(x) : finds the smallest element greater than 'x' .

All operations should be bounded by O (logn). The data structure you would use is:
   a. Heap
   b. Hash Table
   c. Binary Search Tree
   d. Array

4. For each week of the year, and for each day in the week, values have been kept of the number of cars using the Howrah Bridge (integer values) and the peak smog level (real values) in the central district. What is the most appropriate data structure for storing these data within a program that is written to analyze, for specific days of the week, the relationship between cars using the bridge and peak smog levels?
   a. A file
   b. A multidimensional array
   c. A record of arrays
   d. An array of records

algorithmica

Smart Coding & Interview Series
Top20-Basic Test Series
(Online Test - I)

Section B: (4 x 3 mins) = 12 mins
This section contains questions on code complexity (time & space)

1. Assuming T (1) = constant, the recurrence $T(n) = 9 * T(n/3) + n$ is bounded by
   a. $O(n)$
   b. $O(\log n)$
   c. $O(2^n)$
   d. $O(n^2)$

2. Assume that the running time of the function call mystery (n) is O(n).

```
def recursiveFoo(n):
    if(n == 1):
        print("Done)
    else:
        recursiveFoo(n//2)
        recursiveFoo(n//2)
        mystery(n)
```

What is the complexity of this code?
   a. $O(n \log (n))$
   b. $O(n^2)$
   c. $O(n^2 \log n)$
   d. $O(n (\log n)^2)$

3. Consider the following possible data structures for a set of n distinct integers:
   I. A min-heap
   II. An array of length n sorted in increasing order
   III. A balanced binary search tree

For which of these data structures is the time needed to find the 7th largest element O(log n) in the worst case?
   a. I only
   b. I and II
   c. I and III
   d. II and III

4. Assume that this program sorts the numbers between start and end inclusive in the given array. Also assume that there is hardwareSort method will sort an array

whose length is atmost 3 in $\theta(1)$ time and threeWayMerge will merge the given sorted array in $\theta(n)$ time, n being the length of the array. Using these facts, the time complexity of this sort function is,

```
def sort(array, start, end) :
        length = end – start  + 1

        if (length <= 3):
                hardwareSort(array, start, end)

        oneThird = length // 3        # floor of length / 3
        twoThird = 2 * length // 3    # floor of (2 * length) / 3

        sort(array, start, start + oneThird - 1)
        sort(array, start + oneThird,  start + twoThird - 1)
        sort(array, start + twoThird, end)

        threeWayMerge(array, start, end)
```

a. $O(n \log n)$
b. $\theta(n \log_3 n - 2n/3)$ {Read as *log n to the base 3*}
c. $\theta(n \log_3 n)$ {Read as *log n to the base 3*}
d. $O(n^3)$

algorithmica

Smart Coding & Interview Series
Top20-Basic Test Series
(Online Test - I)

Section C: (4 x 4 mins) = 16 mins
In this section, you will be given a code snippet and you are expected to find the find the bugs if any in that snippet.

1. This simple function is intended to multiply two numbers. The two numbers are input in the form of array of its digits. For eg. 45 is input as {4, 5}. table is a 2 D array whose entries are the product of its indices. For eg. table[3][4] is 12. The indices can range from 0 to 9 inclusive. Changing which of the line(s) of this code will make this function meet its requirement.

```
def multiply (a, b, aLen, bLen):
1.      product = 1
2.      for j in range(bLen–1, -1):
3.
4.          for i in range(aLen–1, -1):
5.
6.              product += table[ b[j] ][ a[i] ] * pow(10, aLen – i + 1)
7.
8.      return product
```

I. line 1 to int product = 0;
II. line 6 to table[b[i]][a[j]] from table[b[j]][a[i]]
III. line 6 to pow(10, aLen – i –1) from pow(10, aLen – i + 1)

a. I only
b. I and III
c. I and II
d. III only

2. This function should return the index of the first zero in the input array. If there are no zeros it returns –1.

```
def indexOfZero(array):
    i = 0
    while(arra[i] != 0 and i < len(array)):
        i += 1
    if( i == len(array)):
        return -1
    return i
```

This function will fail when

a. The input array has two 0s.
b. The input array has all 0s.
c. The input array has no 0s.
d. Won't fail in any case.

Smart Coding & Interview Series
Top20-Basic Test Series
(Online Test - I)

3. The following code is supposed to check if a given array of integers is sorted or not. Point out the line that has the error, if there is one.

```
1       def isSorted(data, n):
2
3               if(n == 1):
4                       return true
5               else:
6
7                       temp = isSorted(data, n-1)
8
9                       return (temp) and (data[n-1] <= data[n])
```

a. Line No:3
b. Line No:7
c. Line No:9
d. There is no error

4. Bob is asked to write a function compact that takes a sorted integer array (may contain duplicates) as argument and removes all the duplicate integer entries from the array. He writes the following function:

```
1. def compact(A, len):
2.
3.      insert = 1
4.
5.      for current in range(1, len):
6.
7.            if (A[current] != A[insert-1]):
8.                  A[insert] = A[current]
9.                  insert += 1
10.                 current += 1
11.           else:
12.                 current += 1
13.
```

What needs to be done for the above piece of code to work correctly?

(a) Nothing needs to be done
(b) Change the condition in line number 7 to A[current] != A[insert]
(c) Remove lines 10 and 12
(d) Remove line 12 only

Smart Coding & Interview Series
Top20-Basic Test Series
(Online Test - I)

Section D: (3 questions) = 40 mins
In this section, you will be given a problem and you are expected to find the best possible algorithm and also code the solution.

1. Consider the following class structure:

```
class ball():
    def __init__(self, colour, number):
        self.colour = colour
        self.number = number
```

In an array of size 2n containing n red balls and n blue balls, each ball of given color has a unique number from 1 to n. The balls are arranged in the array such that for any given no. A red ball with that no. will occur before the corresponding blue ball. The distance for any unique number is defined as the difference between the indices of the red ball and blue ball having that number. Find the total distance for the balls as efficiently as you can.

2. Given an array of random numbers, find an efficient algorithm/code to find the largest preceding number for each number. What are time and space complexities of your algorithm.

3. You are given a ternary tree, (each node having 0/3 children) with all the leaf nodes labeled as either 0 or 1, and all non-leaf nodes labeled as -1. Find an efficient algorithm to compute the majority label of the root given the pointer to root of tree. The majority label of a node is the label which the majority of its children are having. What are time and space complexities of your algorithm.