INDIAN STATISTICAL INSTITUTE
CHENNAI CENTRE
M. Stat. (NB Stream) – Semester II
2014–2016
Programming and Data Structures
Mid-term Examination

Total Marks: 44 Maximum Marks: 40

Date: 23 February 2014 **Duration:** 3 hours

1.

(a) Define the terms Data Type, Data Structure, Abstract Data Type, Algorithm. [4]

(b) Order the following functions by growth rate (where the logarithm is to the base 2).
(*i*) n^3 (*ii*) $2^{\log\log n}$ (*iii*) $n^{\log n}$ (*iv*) $2^{\log^3 n}$ (*v*) $n^2 \log^{100} n$ [3]

(c) Consider the following recurrence, with $T(1) = 1$.

$$T(n) = \sqrt{n}\,T(\sqrt{n}) + n$$

Derive a tight upper bound for $T(n)$ in terms of Big-Oh notation. [3]

2.

Give an efficient algorithm to compute $a^{n!} \pmod m$ for positive integers a, n and m. Estimate the time complexity of your algorithm (you may assume that multiplication of two integers modulo m takes 1 unit of time). [6]

3.

An array contains the elements as shown below.

27 31 55 100 6 88

Considering the three sorting algorithms: Selection, Insertion and Bubble, write the state of the array after each pass (one iteration of the outer loop) of the algorithm. [6]

4.

A bag contains n coins and there is at most one counterfeit coin in the bag. The weight of counterfeit coin is different from that of a normal coin. We need to find the counterfeit coin, if it exists, using a two-pan weighing machine (where we can only compare the weights).
(*i*) How many comparisons are needed when $n = 3$. [1]

(*ii*) Give an efficient method to find the counterfeit coin if it exists ($n > 3$) and justify its correctness. Analyse the number of comparisons needed in your method. [5]

5.

Give an efficient algorithm to multiply two upper triangular matrices A and B of size $n \times n$. Count the number of scalar multiplications and additions required in your algorithm. [4]

6.

What is ptr in the following declaration.

```
int (*(*ptr)[2])();
```

[1]

7.

Declare the following statement: a pointer to a function which receives a char pointer and returns an int pointer. [1]

1. *Identify errors, if any, in the following programs.*
2. *Write the output if the program compiles without any error.*
3. *Each question carries 1 mark.*

8.

```
#include<stdio.h>
int main(){
  int i=10;
  i=!i>14;
  printf("i=%d",i);
}
```

9.

```
#include <stdio.h>
#define a 10+20
int main(){
  printf("%d\n",10+2*a);
}
```

10.

```
#include<stdio.h>
int main(){
    char str[30] = "Hello, How are you";
    char *p=str;
```

```
    *(p+4)='\0';
    printf("%s\n", str);
    return 0;
}
```

11.

```
#include<stdio.h>
int main(){
    int a[] = {10, 20, 30, 40, 50};
    int j;
    for(j=0; j<5; j++)
    {
        printf("%d\n", a[j]);
        a++;
    }
    return 0;
}
```

12.

```
#include<stdio.h>
int main(){
    char *str;
    str = "%c\n";
    printf(str, *str);
    return 0;
}
```

13.

```
#include<stdio.h>
int main(){
    char str1[] = "ISI";
    char str2[] = "Chennai";
    char *s1 = str1, *s2 = str2;
    while(*s1++)
       *s2++=*s1;
    printf("%s\n",str2);
    return 0;
}
```

14.

```
#include<stdio.h>
int main(){
    int arr[2][3] = {12, 13, 14, 15, 16,17};
    printf("%d, %d, %d\n", sizeof(arr), sizeof(*arr), sizeof(**arr));
    return 0;
}
```

15.

```
#include<stdio.h>
int main(){
  char not;
  not=!2+300;
  printf("%d\n",not);
}
```

16.

```
#include<stdio.h>
int main(){
  const int i=4;
  float j;
  j = (float ) ++i;
  printf("%d  %f", i,++j);
}
```

17.

```
#include<stdio.h>
#define PRINT "%d %d\n",MAX,MIN
int main(){
  int MAX = 100, MIN =10;
  printf(PRINT+MAX);
}
```