Course Title: Programming Fundamental I
Course Code: CSE-1103
1st Semester, 1st Year, Session: 2016-17

Time: 3 hours **Marks:** 60

Answer any five Questions from the followings. Parts of the same question should be answered consecutively.

1. a) List and explain different features of C language. [3]
   b) Explain the importance of C as a programming language. [3]
   c) Compare and contrast the execution process of an interpreter versus the execution process of a compiler. What implications does interpretation have on performance, if any? [3]
   d) State the algorithm and flowchart to take marks of 6 different subjects and decide whether passed or failed. If the average mark of these subjects is greater than 59 then display passed otherwise failed. [3]

2. a) Briefly describe the structure of a C program with example. [3]
   b) Briefly explain the compiling and running process of a C program. You may take help of a flow chart. [3]
   c) Differentiate between compilation errors and runtime errors. [2]
   d) Why do we need to use comments in programs? [2]
   e) What will be printed by the following program? Why? [2]

```
#include <stdio.h>
int main(){
    float a;
    a = 5/2;
    printf("%0.1f\n",a);
    return 0;
}
```

3. a) Briefly explain the four basic data types in C language. [3]
   b) State the rules for naming *variable* in C language. Which of the following are invalid *variable* name and why? [2+1]
   i) *floats*, ii) *2nd_column*, iii) *x1-x2*, iv) *first name*
   c) Define *Keywords*. List fifteen *Keywords* which are used in C language. [3]
   d) Write a program to print the following output using for loops. [3]

```
*
* *
* * *
* * * *
* * * * *
* * * * * *
```

4. a) What do you mean by operator and operand? Describe different types of operators in C language with example. [1+3]
   b) For the following code, give the values that are printed out by each *printf* statement. Explain your answer. [2]

```
int t; /* This variable is global */
{
    int t = 2;
    printf("%d\n", t);
    {
        printf("%d\n", t);
        t = 3;
    }
    printf("%d\n", t);
}
{
    printf("%d\n", t);
}
```

c) What will be printed by the following code? Why? [2]

```
#include <stdio.h>
int main(){
    int i;
    for(i=0;i<5;i++){
        printf("%c",'a'+i);
    }
    return 0;
}
```

d) All of you are familiar with Mina and Raju. They have decided to play a mathematical game. They are asking mathematical question to each other. All on a sudden, Mina asks Raju whether the integer number *N* is a prime number or not. Raju fails to answer the question. Your task is to help Raju to find the solution. Write a C program to decide whether a given integer number *N* is prime or not a prime. N. B. Mina can ask any number to Raju. [4]

5. a) What do you know about int and unsigned int? Why we use unsigned int? [2]

b) Draw the flowchart for if and if-else statements. [2]

c) Briefly discuss about different types of instructions supported in C. [4]

d) Consider the following program. What will be the value of each variable at the end of program? (Show all the changes in the corresponding lines) [2]

```
void main()
{
    int a=54,b=8;
    if(a>b)
        b++;
    if(b==9)
        --a;
    if(a!=45 && b!=9)
        a=32;
    if(a!=12)
        b=32;
}
```

e) Consider the following program segment. What will be the value of each variable at the end of program? [2]

```
int a, b,c;
a=12;
b=a++;
printf("a=%d, b=%d\n",a,b);
c=++a;
printf("a=%d, c=%d\n",a,c);
```

6. a) How does a structure differ from an array? [2]

b) Find the output of the following C program (Show all the changes in the corresponding lines). [2]

```
#include<stdio.h>
int main()
{
  int a[5] = {4, 1, 9, 16, 15};
  int i, j, m;
  i = ++a[1];
  j = a[1]++;
  m = a[i++];
  printf("%d, %d, %d", i, j, m);
  return 0;
}
```

c) Identify and correct the errors: `char *str = "happy", str[1] = "e", str[2] = "f";` [1]

d) Implement a C program to subtract matrix *B[3][4]* from matrix *A[3][4]*. [3]

e) Implement a C program to find the *reverse* of a string *str* and check whether it is PALINDROME or NOT A PALINDROME. Example: *str*: **level**, reverse: **level**, PALINDROME. [4]

7. a) Describe typical application of pointers in developing program. [2]

b) Distinguish between *(*m)[5]* and *m[5]*. [2]

c) What is the output of the following segment? Why? [2]

```
int m[2];
*(m+1) = 100;
*m = *(m+1);
printf("%d",m[0]);
```

d) What is the significance of EOF? How does an append mode differ from a write mode? [4]

e) What is the principal difference between the functions *malloc* and *calloc*? [2]

8 a) What do you mean by function prototype and function definition? [2]

b) What do you know about call by value and call by reference of a function? [3]

c) What do you meant by actual and formal parameter? Give a simple example. [3]

d) Define recursive function. Applying recursive function, write a program to calculate the factorial of an integer number. [4]