Set A

Birla Institute of Technology & Science, Pilani
First Semester 2022-23
CS F111 – Computer Programming
Mid Semester Examination

===
04/01/2022   **Max. Marks: 50M**   **Duration: 90 mins**
===
ID No: **Name:**
===

Instructions:
- Write the answers only in the space provided. Don't let your answers flow outside the boxes.
- **The marking is binary. No marks will be given for partially correct answers.**
- Over-written answers of any kind will not be accepted for rechecks.
- Wherever there is a semi colon against a blank, you are supposed to write only one C statement. No marks will be awarded wherever multiple statements are written for such questions.

===
Recheck request (Write question number only)

1. Let R1 and R2 be two 4-bit integers in 2's complement form. For the operation R1 + R2, which one of the following values of R1 and R2 gives an arithmetic overflow? Justify your answer by working out each of the following scenarios. **2+2 = 4M**
   A. R1 = 0011, R2 = 0110   B. R1 = 1001, R2 = 1011

A.

B.

2. Convert 87.75 in IEEE 32-bit Floating Point Representation & encode it to hexadecimal. Work out all the steps. **4M**

3. What are three different levels of programming languages? **3M**

4. The compilation stage that gives machine code is _____ **1M**

5. Draw a flowchart for checking if a number *n* entered by the user is a composite number or not. **5M**

6. What is the output of the following program?

```c
#include<stdio.h>
int  Fun( int n){
     if(n%100 == n/100) return 1;
     else return 0;
}

int main(){
     int i,n=2020;
     if(Fun(n)) printf("Yes");
     else printf("No");
     return 0;
}
```
Output:_____ **2M**

7. Given an incomplete C program that calculates the row-wise sum of a 2-D matrix (**a**). At the same time, the program transforms the matrix **a** to now contain only the row-wise sums along its main diagonal, and resets the other elements to zero. For example, if **a={{1,2,3}, {4,5,6}, {7,8,9}}**, then at the end of execution of the function sum, **a** becomes **{{6,0,0}, {0,15,0}, {0,0,24}}**. Complete the implementation by filling in the blanks. Don't declare any new variables. **5x1=5M**

```c
#include <stdio.h>
#define Row 3
#define Col 3
void sum (int a[][])
{
    int i,j;
    for (int i=0; i<_____; i++)
    {
        int sum=0;
        for (int j=0; j<_____; j++)
        {
            _____;

            _____;

            if(i==j) _____;
        }
    }
}

int main(){
    int a[Row][Col]={1,2,3,4,5,6,7,8,9};
    sum(a);
    return 0;
}
```

8. Given an incomplete program that finds and prints unique elements from an integer array. Fill in the blanks to complete the program. Don't declare any new variables. **2x3=6M**

```c
#include <stdio.h>
void printUniqueNumbers(int a[], int n)
{
    int i, j;
    for (i=0; i<n; i++)
    {
        for (j=0; _____; j++){
            if(_____)
                break;

            if(_____)
                printf("%d", a[i]);

        }
    }
}

int main()
{
    int arr[5] = {5, 6, 10, 6, 5};
    printUniqueNumbers(arr,5);
    return 0;
}
```

Consider the following program for **Questions 9 to 11**

```
int x;
static int y;
int main() {
   const int m=10;
   static char z; z = 'A'; z++;
}
```

9. Storage class of variable **x** is_____. It resides in _____ segment of the memory. Its

   default value is _____. **3M**

10. Storage class of variable **m** is _____. It resides in _____ segment of the memory. **2M**

11. How many bytes does **z** occupy? Ans:_____ **1M**

12. Complete the following C program that adds two distances (given in **inches** and **feet**). Two variables **dist1**, **dist2** of **type struct Distance** are given to store the input distances, which can be assumed to be captured in the main() function (not explicitly given below). The function add_dist() is called in the main() function to add distances **dist1** and **dist2** and print their **sum**. Don't declare any additional variable. **4x2 = 8M**

```
struct Distance {
   int feet;
   int inch;
} dist1, dist2, sum;
void add_dist()
{
   // Code to add the two distances

   _____;

   _____;


   // Code to convert inches to feet if greater than 12
   while (sum.inch >= 12) {

      _____;

      _____;
   }

   printf("\nSum of distances = %d feet,%d inches", sum.feet, sum.inch);
}
```

13. Following is an incomplete code of a function that finds the factorial of a number **num**. The factorial computed must be stored in the variable **fact**. Assume that an appropriate function call is made in the main function for the same. Fill in the blanks to complete the code. Don't declare any additional variables. **3x2 = 6M**

```
void findFact(int num, int *fact)
{
   int i;

   _____;

   for(i=1;_____;i++)

      _____;
}
```