**Birla Institute of Technology & Science, Pilani**
**First Semester 2023-24**
**CS F111 – Computer Programming**
**Midsemester Examination**

=====================================================================================

**12/10/2023** **Max. Marks: 60M** **Duration: 90 minutes**

=====================================================================================

**ID No:** **Name:**

=====================================================================================

**Instructions:**

**Invigilator's Signature:**

- This is an in-built question paper. Write the answers only in the space provided.
- Don't let your answers flow outside the boxes/blanks.
- **Over-written answers of any kind will not be accepted for rechecks.**
- Assume that the necessary standard libraries, and the wrapper (main) functions already exist wherever required.

=====================================================================================

| Recheck requests (write in bullets for each question) | Remarks |
|---|---|
| | |

1. Convert -123.6875 into 32-bit IEEE 754 floating point representation. Show all the steps and write the final result in 32-bit binary as well as its equivalent Hexadecimal representation. **9M**

2. Perform binary subtraction of 117 from 86, representing them as 8-bit binary numbers in 2's complement form. Also show the conversion of the result into decimal. **5M**

3. Complete the following C code that finally calculates the LCM of any two numbers ***num1*** and ***num2*** using their HCF. HCF is the highest common factor, while LCM is the Least common multiple of any two numbers. For example, if the two numbers are 75 and 12, their HCF is 3 while their LCM is 300. **7M**

```
int i, j, num1, num2, hcf=1, lcm=0;   //Use only these variables, you are not allowed to declare extra variables
printf("Input the two numbers: \n");
scanf("%d, %d", &num1, &num2);
j = (num1<num2) ? num1 : num2;

//Write code to find the HCF below. Store the value of HCF in hcf variable. Hint: Use while loop

while(_____________________) {

_______________________________________________

_______________________________________________

_______________________________________________

}

// We know that multiplication of HCF and LCM is equivalent to the multiplication of these two numbers.
lcm=( num1* num2)/hcf;
printf("The LCM of %d and %d is : %d\n", num1, num2, lcm);
```

4. Fill in the following incomplete Selection Sort function that sorts an array ***arr[]*** in the descending order. Assume that the wrapper (main) function already exists which calls the function. **(9+4 = 13M)**

```
void selectionSortDescending(int arr[], int size) {
    int i=0, j=0, maxIndex, temp;   //Use only these variables, you are not allowed to declare any extra variables
    //Identify the index of the maximum element to be swapped below

      for (________________________________) {

        ________________________________________

        ________________________________________

        ________________________________________

        ________________________________________

        ________________________________________
        ________________________________________

        //Swap the maximum element using maxIndex obtained above with the current array element

        if (____________________________) {

             ________________________________________

             ________________________________________

             ________________________________________

        }
}
```

5. Complete the following C program to check whether a number is a Strong Number or not. If the sum of factorials of the individual digits of a number is equal to the original number, then it is a Strong Number. For example, the number 145 is a strong number since 1!+4!+5! = 145. **9M**

```
int i, j, num, sum=0;  //Use only these variables, you are not allowed to declare any extra variables
long fact;             //Use only these variables, you are not allowed to declare any extra variables

printf("Input a number to check whether it is a Strong number: ");
scanf("%d", &num);

//Code to find the sum of factorials and store in sum variable. Hint: Using only for loops below

    for (_______________________________________) {

             ________________________________________

             ________________________________________

             ________________________________________

             ________________________________________

             ________________________________________

    }

if(sum==num)
    printf("%d is a Strong number.\n", num);
else
    printf("\%d is not a Strong number.\n", num);
```

6. Complete the following program that checks if the given string is a palindrome or not. It includes ***<stdbool.h>*** to use ***bool*** data type which takes only two values false (0) or true (1). Both the functions *isPalRec() and isPalRec()* return bool values, and not integer values. **(2+2+4+3+6 = 17M)**

```
#include <string.h>
#include <stdbool.h>
bool isPalRec(char str[], int start, int end) { //This is a recursive function
    // If there is only one character in the string

    ______________________

    ______________________
    // If the first and the last characters in the string do not match

    ______________________

    ______________________

    // Recursive function call

    ______________________

    ______________________

   return true;
}

bool isPalindrome(char str[]) {
int n = strlen(str);
// An empty string is considered as palindrome
if (n == 0)
    return true;
return isPalRec(str, 0, n - 1);
}

int main() {
    char str[50];
    //Read the above char array using fgets() function and format it into a string before passing to
isPalindrome() function below

    if (fgets(______________________________) != NULL) {

        __________________________________________

        __________________________________________

        __________________________________________

        __________________________________________

        __________________________________________

        __________________________________________

    }

    if (isPalindrome(str))
        printf("Yes");
    else
        printf("No");
    return 0;
}
```

**************************************************************************************