Roll Number	Section	
Name		

Department of Computer Science and Engineering, IIT Kharagpur

Programming and Data Structure (CS10001)

Class Test 2

Autumn Semester 2019-20	Date: 30/10/2019
Marks: 40	Time: 1 hour

Answer the questions in the spaces provided on the question booklet. For rough work, you may use the extra pages provided in this booklet for the purpose. No other supplementary sheets will be provided.

Question	Q1	Q2	Q3	Q4	TOTAL
Marks					

Q1. Answer the foll	owing.
---------------------	--------

$$(3+2+(3+2) = 10 \text{ marks})$$

(a) Perform the following subtraction operation using *addition*, using 8-bit 2's complement method: $(58)_{10} - (49)_{10}$

You need to show the complete addition operation, and the answer needs to be in 8-bit 2's complement form.

(b) Represent the following decimal numbers using 8-bit sign-magnitude representation:

- (c) Perform the following conversions:
 - (i) $(167.375)_{10} \rightarrow (?)_2$
 - (ii) $(1101011011.01101)_2 \rightarrow (?)_{16}$

(i)			
(ii)			

```
#include <stdio.h>
#include <stdlib.h>
struct point {
   int x, y; };
typedef struct point _PNT;
// *** Function declaration follows ***
                  allocate_array_of_points (int N)
                            /* Data type returned by the function: (1 mark) */
{
       PNT *tmpArray;
       tmpArray = _____ malloc (N * _____
                          /* Allocate memory for N points: (1+1=2 marks) */
       if (tmpArray == _____) return NULL;
                           /* Memory allocation failed: (1 mark) */
                           /* Return address of allocated array of points: (1 mark) */
}
main()
{
       PNT *p;
       int N,i;
       printf ("Give number of points \n");
       scanf ("%d", &N);
       p = ____
                           /* Call function to allocate array of N points: (1 mark) */
       printf ("Read points \n");
       for (i=0;i<N; i++)
         scanf ("%d %d",_____, _
                           /* Read points in the array allocated: 1+1=2 marks */
       for(i=0;i<N; i++)
         printf ("(%d,%d) \n",_____
                           /* Print points read in the same order:1+1=2 marks */
}
```

Q3. Answer the following.

(2+3+5 = 10 marks)

- (a) Choose the correct option in the question below.
 - The function fgetc() returns EOF when
 - (A) End of file is reached
 - (B) When fgetc() fails to read a character
 - (C) Both (A) and (B)
 - (D) None of the above
- (b) Identify the errors, if any, in the following program.

```
#include <stdio.h>
void main()
{
    unsigned char;
    FILE *fp;
    fp = fopen ("trial", 'r');
    while ((ch=getc(fp)) != EOF)
     printf ("%c", ch);
   fclose (*fp);
}
```

Correct option:

(c) Fill in the blanks in the following C program that compares two text files character by character. #include <stdio.h> #include <stdlib.h> int compareFile (FILE *fPtr1, FILE *fPtr2, int *line, int *col); int main() FILE *fPtr1, *fPtr2; // Declare the two file pointers char name1[100], name2[100]; int diff, line, col; /* Input names of the files to compare */ printf ("\nEnter name of first file: "); scanf ("%s", name1); printf ("\nEnter name of second file: "); scanf ("%s", name2); /* Open the two files to compare */ fPtr1 = fopen (name1, "r"); fPtr2 = fopen (name2, "r"); printf ("\nUnable to open file."); exit (-1); /* Call function to compare the files */ diff = compareFile (fPtr1, fPtr2, &line, &col); if (diff == 0)printf ("\nBoth files are equal."); else { printf ("\nFiles are not equal.\n"); printf ("Line: %d, col: %d\n", line, col); fclose(fPtr1); fclose(fPtr2); return 0; /** Function to compare two files. It returns 0 if both files are same, otherwise it returns -1 and sets "line" and "col" to indicate where the files differ. **/ int compareFile (FILE *fPtr1, FILE *fPtr2, int *line, int *col) char ch1, ch2; *line = 1; *col = ____; do { ch1 = fgetc(fPtr1); ch2 = fgetc(fPtr2); if (ch1 == '\n') { // Increment line *line += 1; *col = 1;} /* If characters are not same then return -1 */ if (ch1 != ch2) return ____; *col } while (ch1 != EOF && ch2 != EOF);

return -1;

/* If both files have reached end */

if (_ else

}

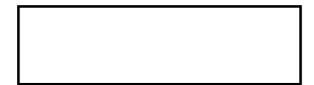
(a) For the following C program, fill up the blank spaces appropriately.

```
#include <stdio.h>
#include <stdlib.h>
#define row 4
#define col 5
int main()
 int i, j;
 int *arr1[row], **arr2 ;
 arr2 = (______) malloc(row * sizeof(int *)); // TYPE CASTING
 for (i=0; i<row; i++)</pre>
       arr2[i] = (int *) malloc(_____
                               // MEMORY SIZE TO BE ALLOCATED
 for (i = 0; i < row; i++)
    for (j = 0; j < col; j++)
       * (_____) = i*j; // STORE IN (i,j)-th ELEMENT OF arr1
 arr2 = arr1;
 for (i = 0; i < row; i++)
    for (j = 0; j < col; j++)
       printf("%d ", arr2[i][j]);
}
```

(b) With respect to the variable declarations as in Q4(a), write the equivalent expressions for the following using "arr1[i][j]": (e.g. *(arr1[i]+j) → arr1[i][j])

*(arr1 + i) + j	
arr1[i] + j	
(*(arr1 + i))[j]	

(c) Consider the declaration "int arr[4][5]", where the address of arr[0][0] in hexadecimal is (64)₁₆. What will the address of arr[1][2] in hexadecimal? Assume that sizeof(int) = 4.



ROUGH WORK

ROUGH WORK