INDIAN INSTITUTE OF TECHNOLOGY KHARAGPUR

	Stamp / Signature of the Invigilator

EXAMINATION (Class Test 2)	SEMESTER (Autumn)

Roll Number									Section		Name	

Subject Number	C	S	1	0	0	0	1	Subject Name	*Programming and Data Structures*

Department / Center of the Student	

Question Number	1	2	3	4	Total Marks
Marks Obtained					

Please write the answers within the boxes provided or fill up the marked blanks in questions 1 through 4. Any answer written elsewhere will not be evaluated. You can use the empty spaces for rough works.

1. Answer the following questions.

    (a) Convert the decimal number 93.671875 into its equivalent binary and hexadecimal numbers. **[0.5+0.5=1]**

Binary: 1011101.101011	Hexadecimal: 5D.AC

    (b) Find out the range of signed integer numbers if it is represented by 6-bit 2's complement notation. **[0.5]**

    ```
    -32 to 31 [(-2)^(5) to 2^(5)-1]
    ```

    (c) Carry out the addition of two 8-bit 2's complement numbers -13 and -11. **[0.5+0.5+0.5 = 1.5]**

2's complement of - 13 =	11110011
2's complement of - 11 =	11110101
Sum of -13 and -11 using 2's complement notation =	11101000

    (d) For the given decimal number -345.25 compute the equivalent IEEE-754 floating point number. **[2]**

```
1 bit sign = 1

8 bit exponent = 10000111

23 bit fractional part = 01011001010000000000000
```

2. Consider that an employee details are being stored through a C structure that contains the name of the employee, his/her employee ID and the salary. Consider the following C program that prints the employee names whose salary is less than a target salary. Complete the following program. You can use the function strlen(const char *str) to find the length of a string. [0.5x10 = 5]

```c
#include<stdio.h>
#include<stdlib.h>
#include<string.h>
typedef struct{
        char* name;
        int id;
        int salary;
} employee;
int main() {
        employee* emp; char tname[50];
        int n,i,tsal;
        printf("\nEnter the number of Employees:");
        scanf("%d",&n);

        emp = (employee*)malloc(n*sizeof(employee));

        printf("\nEnter employee details:");
        for(i=0; i<n; i++) {
                printf("\nEmployee %d:",i+1);
                printf("\nEnter Name:");
                scanf("%s",tname);

                (emp+i)->name=(char*)malloc((strlen(tname)+1)*sizeof(char));
                strcpy((emp+i)->name,tname);

                printf("\nEnter ID:");

                scanf("%d",&(emp+i)->id);

                printf("\nEnter Salary:");

                scanf("%d",&(emp+i)->salary);
        }

        printf("\nEnter target salary:");
        scanf("%d",&tsal);
        printf("\n");
        for(i=0; i<n; i++) {

                if((emp+i)->salary < tsal){

                        printf("%s\n",(emp+i)->name);
                }
        }

        for(i=0; i<n; i++) {
```

```
        free((emp+i)->name);
    }

    free(emp);
    return 0;
}
```

3. Given an n produce a two dimensional array where the number of columns in row will increase by one till mid-point and then decrease. The first row will have 1 column. Also all the cells need to be filled with natural numbers in increasing order. The following are the output for $n = 4$ and $n = 5$ respectively.

```
0
1 2
3 4
5
```

```
0
1 2
3 4 5
6 7
8
```

Complete the code to produce the output for a given n.   **[0.5x10 = 5]**

```
#include<stdio.h>
#include<stdlib.h>
int main(){
        int **p;
        int i,k,j, n, count = 0;
        scanf("%d",&n);

        p = (int **)malloc(n * sizeof(int *));
        for(i = 0; i < n; i ++){

                if (i < n/2)
                        k = i + 1;
                else
                        k = n - i;

                p[i] = (int *)malloc(k * sizeof(int));

                for(j = 0; j < k; j++,count++)
                        p[i][j] = count;
                for(j = 0; j < k; j++)
                        printf("%d ", p[i][j]);
                printf("\n");
        }
}
```

4. Given the details of n employees (employee name and salary), write a C program which sorts the employee names according to the length (no of characters) of their names. If the lengths of employee names are same, then break the tie by alphabetical order. Finally, display the details of employees in the above mentioned order. Fill up the missing code below to complete the program.   **[1x5 = 5]**

```
#include<stdio.h>
#include<string.h>
#include<stdlib.h>
struct employee{
    char name[50];
    int salary;
};
int main(){
    int i=0,j=0,n,high;
```

```c
    struct employee  t, *emp;
    printf("\n\n Enter the number of employees : ");
    scanf("%d",&n);

    emp = (struct employee*)malloc(n*sizeof(struct employee));
    for(i=0;i<n;i++){
        printf("\nEnter employee-%d Details\n",i+1);
        printf("Enter Name     : ");
        scanf("%s",emp[i].name);
        printf("salary         : ");
        scanf("%d",&(emp[i].salary));
    }
// sort the employees by their name length
    for(i=0;i<n;i++){
        for(j=i+1;j<n;j++){

            if(strlen(emp[i].name)>strlen(emp[j].name)){
                t=emp[i];
                emp[i]=emp[j];
                emp[j]=t;
            }
        // if lengths are same, break the tie by alphabetical order

            else if(strlen(emp[i].name)==strlen(emp[j].name)){

                if(strcmp(emp[i].name,emp[j].name)>0){
                    t=emp[i];
                    emp[i]=emp[j];
                    emp[j]=t;

                }
            }
        }
    }
// printing the employee details in the sorted order
    printf("\n employee details in the sorted order \n");
    for (i=0;i<n;i++){

        printf("\n %s\t%d",emp[i].name,emp[i].salary);
    }
}
```