Assignment – 21

1. Define a structure Employee with member variables id, name, salary.

```c
#include<stdio.h>
#include<string.h>
struct Employee
{
    int id;
    char name[20];
    int salary;
};
int main()
{
    struct Employee e1;
    e1.id = 1;
    strcpy(e1.name,"Alice");
    e1.salary=20000;
    printf("\nid : %d",e1.id);
    printf("\nName : %s",e1.name);
    printf("\nSalary : %d",e1.salary);
    return 0;
}
```

2. Write a function to take input employee data from the user. [Refer structure from question 1]

```c
#include<stdio.h>
#include<string.h>
struct Employee
{
    int id;
    char name[20];
    int salary;
}e1;

void takeinput()
{
    printf("Enter id : ");
    scanf("%d",&e1.id);
    printf("Enter Name : ");
    scanf("%s",e1.name);
    printf("Enter salary : ");
    scanf("%d",&e1.salary);
}
int main()
{
    takeinput();
    return 0;
}
```

3. Write a function to display employee data. [Refer structure from question 1]

```c
#include<stdio.h>
#include<string.h>
struct Employee
{
    int id;
```

```
    char name[20];
    int salary;
}e1;

void takeinput()
{
    printf("Enter id : ");
    scanf("%d",&e1.id);
    printf("Enter Name : ");
    scanf("%s",&e1.name);
    printf("Enter salary : ");
    scanf("%d",&e1.salary);
}
void display()
{
    printf("\nid : %d",e1.id);
    printf("\nName : %s",e1.name);
    printf("\nSalary : %d",e1.salary);
}
int main()
{
    takeinput();
    display();
    return 0;
}
```

4. Write a function to find the highest salary employee from a given array of 10 employees. [Refer structure from question 1]

```
#include<stdio.h>
#include<string.h>

int max=0;
int flag=0;
struct Employee
{
    int id;
    char name[20];
    int salary;
}e[10];

void maxsalary()
{
    int i;
    for(i=0;i<10;i++)
    {
        if(e[i].salary>max)
        {
            max = e[i].salary;
            flag = i;
        }
    }
}

int main()
{
    e[0].id=1;
```

```
    strcpy(e[0].name,"Aa");
    e[0].salary=1500;
    e[1].id=2;
    strcpy(e[1].name,"Bb");
    e[1].salary=1800;
    e[2].id=3;
    strcpy(e[2].name,"Cc");
    e[2].salary=2500;
    e[3].id=4;
    strcpy(e[3].name,"Dd");
    e[3].salary=500;
    e[4].id=5;
    strcpy(e[4].name,"Ee");
    e[4].salary=1000;
    e[5].id=6;
    strcpy(e[5].name,"Ff");
    e[5].salary=1300;
    e[6].id=7;
    strcpy(e[6].name,"Gg");
    e[6].salary=8500;
    e[7].id=8;
    strcpy(e[7].name,"Hh");
    e[7].salary=7500;
    e[8].id=9;
    strcpy(e[8].name,"Ii");
    e[8].salary=9500;
    e[9].id=10;
    strcpy(e[9].name,"Jj");
    e[9].salary=100;

    maxsalary();
    printf("Highest Salary Employee Information\n");
    printf("id = %d , Name = %s , Salary = %d",e[flag].id,e[flag].name,max);
    return 0;
}
```

5. Write a function to sort employees according to their salaries [refer structure from question 1]

```
#include<stdio.h>
#include<string.h>

struct Employee
{
    int id;
    char name[20];
    int salary;
}e[10];

void sortsalary()
{
    struct Employee temp;
    int i,j;
    for(i=0;i<9;i++)
    {
        for(j=i;j<10;j++)
        {
            if(e[i].salary>e[j].salary)
```

```
            {
                temp = e[i];
                e[i] = e[j];
                e[j] = temp;
            }
        }
    }
}

int main()
{
    int i;
    e[0].id=1;
    strcpy(e[0].name,"Aa");
    e[0].salary=1500;
    e[1].id=2;
    strcpy(e[1].name,"Bb");
    e[1].salary=1800;
    e[2].id=3;
    strcpy(e[2].name,"Cc");
    e[2].salary=2500;
    e[3].id=4;
    strcpy(e[3].name,"Dd");
    e[3].salary=500;
    e[4].id=5;
    strcpy(e[4].name,"Ee");
    e[4].salary=1000;
    e[5].id=6;
    strcpy(e[5].name,"Ff");
    e[5].salary=1300;
    e[6].id=7;
    strcpy(e[6].name,"Gg");
    e[6].salary=8500;
    e[7].id=8;
    strcpy(e[7].name,"Hh");
    e[7].salary=7500;
    e[8].id=9;
    strcpy(e[8].name,"Ii");
    e[8].salary=9500;
    e[9].id=10;
    strcpy(e[9].name,"Jj");
    e[9].salary=100;

    sortsalary();
    printf("<<-- List of Employee in Sorted order -->>\n");
    printf("id  Name   Salary\n");
    for(i=0;i<10;i++)
    {
        printf(" %d   %s    %d",e[i].id,e[i].name,e[i].salary);
        printf("\n");
    }
    return 0;
}
```

6. Write a function to sort employees according to their names [refer structure from question 1]

```c
#include<stdio.h>
#include<string.h>

struct Employee
{
    int id;
    char name[20];
    int salary;
}e[10];

void sortname()
{
    struct Employee temp;
    int i,j;
    for(i=0;i<9;i++)
    {
        for(j=i;j<10;j++)
        {
            if(e[i].name[1]>e[j].name[1])
            {
                temp = e[i];
                e[i] = e[j];
                e[j] = temp;
            }
        }
    }
}

int main()
{
    int i;
    e[0].id=1;
    strcpy(e[0].name,"Aa");
    e[0].salary=1500;
    e[1].id=2;
    strcpy(e[1].name,"Bb");
    e[1].salary=1800;
    e[2].id=3;
    strcpy(e[2].name,"Cc");
    e[2].salary=2500;
    e[3].id=4;
    strcpy(e[3].name,"Dd");
    e[3].salary=500;
    e[4].id=5;
    strcpy(e[4].name,"Ee");
    e[4].salary=1000;
    e[5].id=6;
    strcpy(e[5].name,"Ff");
    e[5].salary=1300;
    e[6].id=7;
    strcpy(e[6].name,"Gg");
    e[6].salary=8500;
    e[7].id=8;
    strcpy(e[7].name,"Hh");
    e[7].salary=7500;
```

```
    e[8].id=9;
    strcpy(e[8].name,"Ii");
    e[8].salary=9500;
    e[9].id=10;
    strcpy(e[9].name,"Jj");
    e[9].salary=100;

    sortname();
    printf("<<-- List of Employee in Sorted order -->>\n");
    printf("id  Name   Salary\n");
    for(i=0;i<10;i++)
    {
        printf(" %d   %s    %d",e[i].id,e[i].name,e[i].salary);
        printf("\n");
    }
    return 0;
}
```

7. Write a program to calculate the difference between two time periods.

```
#include<stdio.h>
struct Time
{
    int hr;
    int min;
    int sec;
}t1,t2;

void difference()
{
    int temp1 = t1.hr*60*60 + t1.min*60 + t1.sec;
    int temp2 = t2.hr*60*60 + t2.min*60 + t2.sec;
    int diff = temp1 - temp2 ;
    printf("%d Hr %d min %d sec",diff/3600,(diff%3600)/60,((diff%3600)%60));
}

int main()
{
    printf("Enter First Time in Hour Minute & Sec : ");
    scanf("%d%d%d",&t1.hr,&t1.min,&t1.sec);

    printf("Enter Seconf Time in Hour Minute & Sec : ");
    scanf("%d%d%d",&t2.hr,&t2.min,&t2.sec);

    difference();
    return 0;
}
```

8. Write a program to store information of 10 students and display them using structure.

```
#include<stdio.h>
#include<string.h>
struct Student
{
    int roll;
    char name[20];
    int marks;
}s[10];
```

```
int main()
{
    int i;
    printf("\n!!  Marks must be out of 100 !! \n\n");
    printf("<<--Enter Information of 10 students-->>\n\n");
    for(i=0;i<10;i++)
    {
        s[i].roll = i+1;
        printf("\nEnter Information For Roll Number (%d) ->",s[i].roll);
        printf("\nEnter Name : ");
        scanf("%s",s[i].name);
        printf("Enter Marks : ");
        scanf("%d",&s[i].marks);
    }

    printf("\n\n<<--Information of 10 students-->>\n");
    for(i=0;i<10;i++)
    {
        printf("\nRoll Number : %d",i+1);
        printf("\nName : %s",s[i].name);
        printf("\nMarks : %d",s[i].marks);
        printf("\n");
    }
    return 0;
}
```

9. Write a program to store information of n students and display them using structure.

```
#include<stdio.h>
#include<string.h>
struct Student
{
    int roll;
    char name[20];
    int marks;
};

int main()
{
    int i,n;
    printf("Enter Number of Students : ");
    scanf("%d",&n);
    printf("\n!!  Marks must be out of 100 !! \n\n");
    struct Student s[n];
    printf("<<--Enter Information of %d students-->>\n\n",n);
    for(i=0;i<n;i++)
    {
        s[i].roll = i+1;
        printf("\nEnter Information For Roll Number (%d) ->",s[i].roll);
        printf("\nEnter Name : ");
        scanf("%s",s[i].name);
        printf("Enter Marks : ");
        scanf("%d",&s[i].marks);
    }

    printf("\n\n<<--Information of %d students-->>\n",n);
```

```
    for(i=0;i<n;i++)
    {
        printf("\nRoll Number : %d",i+1);
        printf("\nName : %s",s[i].name);
        printf("\nMarks : %d",s[i].marks);
        printf("\n");
    }
    return 0;
}
```

10. Write a program to enter the marks of 5 students in Chemistry, Mathematics and Physics (each out of 100) using a structure named Marks having elements roll no., name, chem_ marks, maths_marks and phy_marks and then display the percentage of each student.

```
#include<stdio.h>

struct Marks
{
    int roll;
    char name[20];
    int chem_marks;
    int maths_marks;
    int phy_marks;
    float perc;
}s[5];

void percentage(int i)
{
    float add,result;
    add = s[i].chem_marks + s[i].maths_marks + s[i].phy_marks;
    result = (add/3) ;
    s[i].perc = result;
}

int main()
{
    int i;
    printf("\n!! Marks of Student must be out of 100 !!\n\n");
    printf("<<--Enter Information of 5 students-->>\n");
    for(i=0;i<5;i++)
    {
        s[i].roll = i+1;
        printf("\nEnter Name & Marks of Roll Number (%d) ->\n",s[i].roll);
        printf("Name : ");
        scanf("%s",s[i].name);
        printf("Chemistry marks :");
        scanf("%d",&s[i].chem_marks);
        printf("Maths marks :");
        scanf("%d",&s[i].maths_marks);
        printf("Physics marks :");
        scanf("%d",&s[i].phy_marks);
        printf("\n");
    }

    printf("<<--Percentage of Students-->>\n");
    for(i=0;i<5;i++)
    {
```

```
        percentage(i);
        s[i].roll = i+1;
        printf("\nRoll Number : %d",s[i].roll);
        printf("\nName : %s",s[i].name);
        printf("\nPercentage : %.2f",s[i].perc);
        printf("\n");
    }
    printf("\n");
    return 0;
}
```