Batch: P4-1	**Roll No.: 16010423076**
Experiment / assignment / tutorial No. 7	
Grade: AA / AB / BB / BC / CC / CD /DD	
Signature of the Staff In-charge with date	

TITLE: Write a program in C to demonstrate use of structures and union.

AIM: Write a program to manage an employee database using structure and union in C. Each Employee has the following information:

1. Employee ID(integer)
2. Name(string)
3. Department(string)
4. Salary(float)

You need to implement the following functionalities:

1. Create a structure named Employee with the appropriate data members to store the information mentioned above.
2. Create a union named EmployeeInfo that can hold either the Name or Department information.
3. Write a function addEmployee that takes user input for each employee's information and stores it in an array of structures.
4. Write a function printEmployeeDetails that takes an employee's ID as input and prints all available details for that employee.
5. Write a function updateEmployeeInfo that takes an employee's ID and allows the user to update either the Name or Department information using the EmployeeInfo union.
6. Implement a menu-driven program that allows the user to perform the above operations. Include options to add a new employee, print employee details, update employee information, and exit the program.

Expected OUTCOME of Experiment:

Design modular programs using functions and the use of structure and union(CO4).

Books/ Journals/ Websites referred:

1. Programming in C, second edition, Pradeep Dey and Manas Ghosh, Oxford University Press.
2. Programming in ANSI C, fifth edition, E Balagurusamy, Tata McGraw Hill.
3. Introduction to programming and problem solving , G. Michael Schneider ,Wiley India edition.

Problem Definition:
The program accepts a choice from the user using a switch case statement and generates output accordingly.

Algorithm:

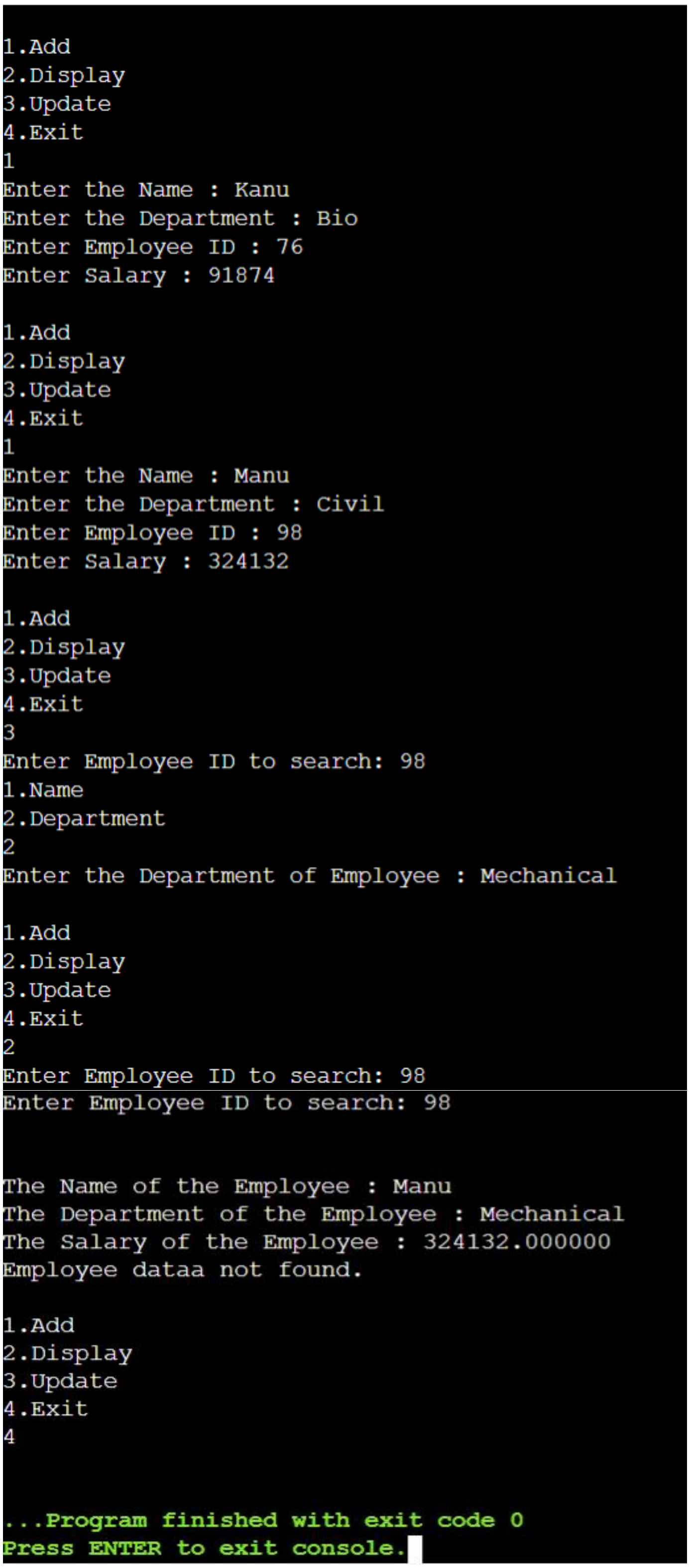
1. Declare necessary variables and structures.
2. Implement functions for adding an employee, printing employee details, and updating employee information.
3. Use a while loop to display a menu-driven interface until the user chooses to exit.
4. Inside the loop, prompt the user to choose an option from the menu (add, display, update, exit).
5. Based on the user's choice, call the corresponding function to perform the desired operation.
6. Ensure appropriate input validation and error handling.
7. Exit the program when the user chooses to exit.

Implementation details:

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

int i = 0;
struct Employee{
int eid;
char name[100];
char department[100];
float salary;
} s[100];

union EmployeeInfo{
char name[100];
char department[100];
};

void addEmployee(){
i++;
printf("Enter the Name : ");
scanf("%s",s[i].name);
printf("Enter the Department : ");
scanf("%s",s[i].department);
printf("Enter Employee ID : ");
scanf("%d",&s[i].eid);
printf("Enter Salary : ");
scanf("%f",&s[i].salary);
}

void printEmployeeDetails(){
int empID;
printf("Enter Employee ID to search: ");
scanf("%d", &empID);
for(int j=1;j<=i;j++){
        if(s[j].eid == empID){
            printf("\n\nThe Name of the Employee : %s",s[j].name);
            printf("\nThe Department of the Employee : %s",s[j].department);
            printf("\nThe Salary of the Employee : %f\n",s[j].salary);
        }
}
}
```

```
void updateEmployeeInfo(){
int empID;
printf("Enter Employee ID to search: ");
scanf("%d",&empID);

for(int j=1;j<=i;j++){
        if(s[j].eid == empID){
            int choice;
            printf("1.Name\n2.Department\n");
            scanf("%d", &choice);
            switch(choice){
                case 1:
                    printf("Enter the Name of Employee : ");
                    scanf("%s", s[j].name);
                    break;
                case 2:
                    printf("Enter the Department of Employee : ");
                    scanf("%s", s[j].department);
                    break;
                default:
                    printf("Error....\n");
                }
        }

}
}




int main(){
int k = 1;
while(k != 0){
        int n;
        printf("\n1.Add\n2.Display\n3.Update\n4.Exit\n");
        scanf("\n\n %d", &n);

        switch(n){
            case 1:
                addEmployee();
                break;
            case 2:
                printEmployeeDetails();
                break;
            case 3:
                updateEmployeeInfo();
                break;
            case 4:
                exit(0);
                break;
            default:
                printf("Error....\n");
                break;
        }
}
return 0;
}
```

Output(s):



```
1.Add
2.Display
3.Update
4.Exit
1
Enter the Name : Kanu
Enter the Department : Bio
Enter Employee ID : 76
Enter Salary : 91874

1.Add
2.Display
3.Update
4.Exit
1
Enter the Name : Manu
Enter the Department : Civil
Enter Employee ID : 98
Enter Salary : 324132

1.Add
2.Display
3.Update
4.Exit
3
Enter Employee ID to search: 98
1.Name
2.Department
2
Enter the Department of Employee : Mechanical

1.Add
2.Display
3.Update
4.Exit
2
Enter Employee ID to search: 98
Enter Employee ID to search: 98


The Name of the Employee : Manu
The Department of the Employee : Mechanical
The Salary of the Employee : 324132.000000
Employee dataa not found.

1.Add
2.Display
3.Update
4.Exit
4


...Program finished with exit code 0
Press ENTER to exit console.
```

Conclusion:
I've learned fundamental concepts in C programming for organizing and manipulating data. Structures allow me to group related data fields under a single entity, facilitating better organization and accessibility of information. Conversely, unions provide a way to utilize a single memory location to store different types of data, optimizing memory usage and enabling the representation of various views of the same data.

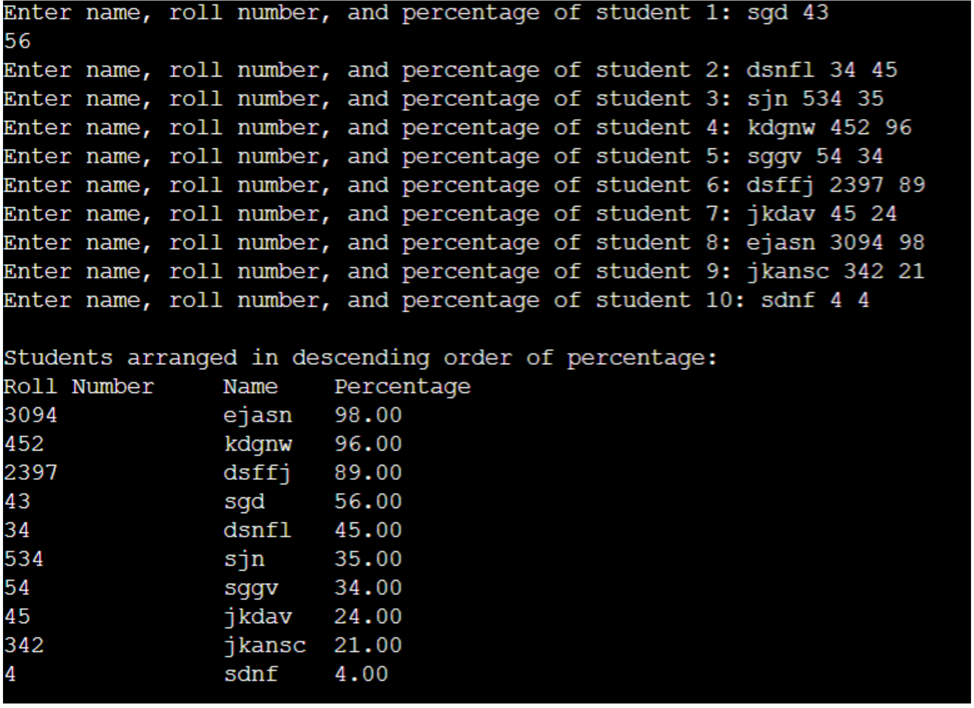
Post Lab Descriptive Questions

- WAP to accept student name, roll number and percentage for 10 students using array of structures and arrange them in descending order of their percentage.
  Input:

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


struct Student {
char name[100];
int rollno;
float percentage;
};

int main() {
struct Student s[10];
for (int i=0;i<10;i++) {
    printf("Enter name,roll number & percentage(separated by space)of student %d: ",i+1);
    scanf("%s %d %f",s[i].name,&s[i].rollno,&s[i].percentage);
}

for (int i=0; i<9;i++) {
    for (int j=0;j<9-i;j++) {
        if (s[j].percentage < s[j+1].percentage) {
        struct Student temp = s[j];
        s[j] = s[j+1];
        s[j+1] = temp;
        }
    }
}
printf("\nThe order of decreasing percentage : \n");
printf("Roll Number\tName\tPercentage\n");
for (int i=0;i<10;i++) {
```

```
printf("%d\t\t%s\t%f\n", s[i].rollno,s[i].name,s[i].percentage);
}
}
```

Output:



```
Enter name, roll number, and percentage of student 1: sgd 43
56
Enter name, roll number, and percentage of student 2: dsnfl 34 45
Enter name, roll number, and percentage of student 3: sjn 534 35
Enter name, roll number, and percentage of student 4: kdgnw 452 96
Enter name, roll number, and percentage of student 5: sggv 54 34
Enter name, roll number, and percentage of student 6: dsffj 2397 89
Enter name, roll number, and percentage of student 7: jkdav 45 24
Enter name, roll number, and percentage of student 8: ejasn 3094 98
Enter name, roll number, and percentage of student 9: jkansc 342 21
Enter name, roll number, and percentage of student 10: sdnf 4 4

Students arranged in descending order of percentage:
Roll Number     Name    Percentage
3094            ejasn   98.00
452             kdgnw   96.00
2397            dsffj   89.00
43              sgd     56.00
34              dsnfl   45.00
534             sjn     35.00
54              sggv    34.00
45              jkdav   24.00
342             jkansc  21.00
4               sdnf    4.00
```

- WAP to display employee name, ID and year of experience using union.

```
#include <stdio.h>

#include <string.h>


union EmployeeInfo {

char name[100];

int employeeID;

int experience;

};
```

```
int main() {
union EmployeeInfo s;
printf("Enter employee name: ");
scanf("%s",s.name);

printf("Enter employee ID: ");
scanf("%d",&s.employeeID);

printf("Enter years of experience: ");
scanf("%d",&s.experience);

printf("\nEmployee Information - \n");
printf("Name: %s\n",s.name);
printf("ID: %d\n",s.employeeID);
printf("Years of Experience: %d\n",s.experience);
}
```

- Virtual lab on Structure and Union

  https://cse02-iiith.vlabs.ac.in/exp/structures/simulation.html

Date: ______________ **Signature of faculty in-charge**