AIM:	Apply the concepts of structures/union to solve a given problem.

Program 1

PROBLEM STATEMENT:	A men's sports club keeps elaborate computerized records of all its members. The records contain typical information such as age, address, etc. of each person. But there is also information about whether a member is an active playing members, about whether he is married, and so on; if he is married the record contains information about his wife's name, the no. of children and their names. Write a program which demonstrates how such a system might be implemented. Show how the names of the wives of all active playing members might be printed.
PROGRAM:	

```c
#include <stdio.h>
//Structure taking information of player
typedef struct player {
    int ages;
    char name[100];
    char address[1000];
    int active;
    int married;
}player_t;
//Structure taking information of player's wife and kids if there is any
typedef struct players_married {
    int wife_ages;
    char wife_name[100];
    int no_of_child;
    char name_of_child[100];
    int age_of_child;
}married_t;
//Function to take player's informaton from user
void read_player(player_t pla[],int n,married_t p[]){
    for (int i=0;i<n;i++){
        printf("Enter the name of the player: ");
        scanf("%s",pla[i].name);
        printf("Enter the address: ");
        scanf(" %[^\n]",pla[i].address);
        printf("Enter the age of the player: ");
        scanf("%d",&pla[i].ages);
        printf("Enter 1 if the player is Active and 0 if he is not: ");
        scanf("%d",&pla[i].active);
        printf("Enter 1 if the player is married and 0 if he is not: ");
        scanf("%d",&pla[i].married);
        if (pla[i].married){
            printf("Enter wife name: ");
            scanf(" %[^\n]",p[i].wife_name);
            printf("Enter the age of wife: ");
            scanf("%d",&p[i].wife_ages);
            printf("Enter the no of kid of player: ");
            scanf("%d",&p[i].no_of_child);
            for (int i=0;i<p[i].no_of_child;i++){
```

```c
            printf("Enter name of child: ");
            scanf(" %[^\n]",p[i].name_of_child);
            printf("Enter age of child: ");
            scanf("%d",&p[i].age_of_child);
        }
    }
  }
}
//Function to print the information of players
void print_player(player_t pla[],int n,married_t p[]){
    for (int i=0;i<n;i++){
        printf("Name of player: %s",pla[i].name);
        printf("\nAge of player: %d",pla[i].ages);
        printf("\nAddress of player: %s",pla[i].address);
        if (pla[i].active){
            printf("\nPlayer is Active");
        }
        else
            printf("\nPlayer is not active");
        if (pla[i].married){
            printf("\nHE IS MARRIED!");
            printf("\nName of wife: %s",p[i].wife_name);
            printf("\nNumber of child: %d",p[i].no_of_child);
            for (int i=0;i<p[i].no_of_child;i++){
                printf("\nName of child: %s",p[i].name_of_child);
                printf("\nEnter age of child: %d",p[i].age_of_child);
            }
        }
    }
}

int main()
{
    int no_of_player;
    printf("Enter the number of player: ");
    scanf("%d",&no_of_player);
    player_t pla[no_of_player];
    married_t p[no_of_player];
    read_player(pla,no_of_player,p);
    print_player(pla,no_of_player,p);
    return 0;
}
```

RESULT:

```
Enter the number of player: 3
Enter the name of the player: Pranay
Enter the address: shanti darshmmn
Enter the age of the player: 34
Enter 1 if the player is Active and 0 if he is not: 1
Enter 1 if the player is married and 0 if he is not: 0
Enter the name of the player: Pran
Enter the address: hdhdnfs
Enter the age of the player: 23
Enter 1 if the player is Active and 0 if he is not: 1
Enter 1 if the player is married and 0 if he is not: 1
Enter wife name: Prdsshi
Enter the age of wife: 23
Enter the no of kid of player: 4
Enter name of child: rohan
Enter age of child: 12
Enter name of child: raju
Enter age of child: 2
Enter name of child: ram
Enter age of child: 4
Enter the name of the player: ramu
Enter the address: jshdjsd
Enter the age of the player: 12
Enter 1 if the player is Active and 0 if he is not: 1
Enter 1 if the player is married and 0 if he is not: 0
------------------------------------------
Name of player: Pranay
Age of player: 34
Address of player: shanti darshmmn
Player is Active
------------------------------------------
------------------------------------------
Name of player: Pran
Age of player: 23
Address of player: hdhdnfs
Player is Active
HE IS MARRIED!
Name of wife: Prdsshi
Number of child: 4
Name of child: rohan
Enter age of child: 12
Name of child: raju
Enter age of child: 2
Name of child: ram
Enter age of child: 4
------------------------------------------
------------------------------------------
Name of player: ramu
Age of player: 12
Address of player: jshdjsd
Player is Active
------------------------------------------
```

Program 2	
PROBLEM STATEMENT :	An airline reservation system maintains records for possible flights consisting of STARTING POINT 3 character code DESTINATION 3 character code STARTING TIME integer on scale 0001 – 2400 ARRIVAL TIME integer on scale 0001 – 2400 SEATS positive integer in suitable range. Your program is to read 20 such records followed by queries of the form STARTING POINT– DESTINATION, one to a line. For each query find whether there is a possible flight with a seat available; if so reduce the number of seats by one and print out the flight details (or an apology).
| **PROGRAM:** | ```c
#include <stdio.h>
#include<string.h>
#include <stdbool.h>
//Structure whcih stores the information of flights
typedef struct airline{
   char STARTING_POINT[4];
   char DESTINATION[4];
   int ARRIVAL_TIME;
   int STARTING_TIME;
   int no_of_seats;
}airline_t;
//Struture which stores the information of query
typedef struct queries{
   char STARTING_POINT[4];
   char DESTINATION[4];
   int book;
}queries_t;
//To read tha no of flight
void read_arline(airline_t f[],int n){
   for (int i=0;i<n;i++){
      printf("Enter the starting point of flight(3 Code): ");
      scanf("%s",f[i].STARTING_POINT);
      printf("Enter the destination of the flight(3 Code): ");
      scanf("%s",f[i].DESTINATION);
      printf("Enter the departure time in hhmm form: ");
      scanf("%d",&f[i].STARTING_TIME);
      printf("Enter the arrival time in hhmm form: ");
      scanf("%d",&f[i].ARRIVAL_TIME);
      printf("Enter the number of seats available: ");
      scanf("%d",&f[i].no_of_seats);
   }
}
//To take query
void queries(airline_t f[],int n,queries_t q){
   bool availability=false;
   printf("=================================================");
   printf("\nQuery");
``` |

```c
        printf("\nEnter the starting point of flight(3 Code): ");
        scanf("%s",q.STARTING_POINT);
        printf("Enter the destination of the flight(3 Code): ");
        scanf("%s",q.DESTINATION);
        for(int i=0;i<n;i++){
            if (strcmp(f[i].STARTING_POINT,q.STARTING_POINT) &&
    strcmp(q.DESTINATION,f[i].DESTINATION)){
                availability=true;
                printf("These are the flight for you:");
                printf("\nDeparture Time: %d",f[i].STARTING_TIME);
                printf("\nArrival Time: %d",f[i].ARRIVAL_TIME);
                printf("\nDo you want to book this flight. Enter 1 for Yes and 0 for No: ");
                scanf("%d",&q.book);
                if(q.book==1 && f[i].no_of_seats!=0){
                    f[i].no_of_seats--;
                    printf("Your flight has been booked");
                }
                else
                    break;
            }
        }
        if (!availability)
            printf("Sorry we do not have any flight");
        printf("\n=============================================");
}
int main()
{
    int no_of_flight;
    airline_t f[no_of_flight];
    queries_t q;
    printf("Enter the number of flight: ");
    scanf("%d",&no_of_flight);
    read_arline(f,no_of_flight);
    queries(f,no_of_flight,q);
    return 0;
}
```

RESULT:

```
Enter the number of flight: 2
Enter the starting point of flight(3 Code): BOR
Enter the destination of the flight(3 Code): VAP
Enter the departure time in hhmm form: 1500
Enter the arrival time in hhmm form: 1600
Enter the number of seats available: 14
Enter the starting point of flight(3 Code): BJP
Enter the destination of the flight(3 Code): CON
Enter the departure time in hhmm form: 1700
Enter the arrival time in hhmm form: 1800
Enter the number of seats available: 3343
=========================================
Query
Enter the starting point of flight(3 Code): BOR
Enter the destination of the flight(3 Code): VAP
These are the flight for you:
Departure Time: 1700
Arrival Time: 1800
Do you want to book this flight. Enter 1 for Yes and 0 for No: 1
Your flight has been booked
=========================================
```

Program 3	
PROBLEM STATEMENT:	Write a program to store the name, matches played and goals scored by 'n' hockey players using structure. generate a list with goals scored in descending order i.e display the output in table form in order of maximum goals scored to minimum goals scored.
PROGRAM:	(see code below)

```c
#include <stdio.h>
//Structure declartion
typedef struct player{
    char name[100];
    int matches_played;
    int goals_scored;
}player_t;
//to take player information from user
void read_player(player_t p[], int no_of_player){
    for(int i=0;i<no_of_player;i++){
        printf("Enter the name of player: ");
        scanf("%s",p[i].name);
        printf("Enter the matches played by %s :",p[i].name);
        scanf("%d",&p[i].matches_played);
        printf("Enter the number of goals scored by %s :",p[i].name);
        scanf("%d",&p[i].goals_scored);
    }
}
//To sort the player
void sort_player(player_t p[],int no_of_player){
    int i,j,min_index;
    player_t t;
    for(i=0;i<no_of_player-1;i++){
        min_index=i;
        for(j=i+1;j<no_of_player;j++){
            if (p[j].goals_scored>p[min_index].goals_scored){
                min_index=j;
            }
        }
        t=p[min_index];
        p[min_index]=p[i];
        p[i]=t;
    }
}
//To print the information of players
void print_player(player_t p[],int no_of_player){
    printf("\n");
    printf("\n-----------------------------------------------------------");
    printf("\nNAMES\tMATCHES\tGOALS");
    printf("\n-----------------------------------------------------------");
    for(int i=0;i<no_of_player;i++){
        printf("\n%s\t%d\t%d",p[i].name,p[i].matches_played,p[i].goals_scored);
    }
}
int main()
```

```
{
    int no_of_player;
    printf("Enter the number of players: ");
    scanf("%d",&no_of_player);
    player_t p[no_of_player];
    read_player(p,no_of_player);
    sort_player(p,no_of_player);
    print_player(p,no_of_player);


    return 0;
}
```

RESULT:

```
Enter the number of players: 4
Enter the name of player: Ronaldo
Enter the matches played by Ronaldo :5
Enter the number of goals scored by Ronaldo :6
Enter the name of player: Messi
Enter the matches played by Messi :5
Enter the number of goals scored by Messi :5

Enter the name of player: Nennar
Enter the matches played by Nennar :5
Enter the number of goals scored by Nennar :2
Enter the name of player: Pranay
Enter the matches played by Pranay :5
Enter the number of goals scored by Pranay :8



------------------------------------------------------
NAMES    MATCHES GOALS
------------------------------------------------------
Pranay   5          8
Ronaldo  5          6
Messi    5          5
Nennar   5          2
```

| **CONCLUSION:** | We learned to apply the concepts of structures/union to solve a given problem. |