Assignment 14

```
//Complex (real, imaginary)
#include<stdio.h>
struct complex{
	int real;
	int img;

};
void main(){

	struct complex c;

	c.real=10;
	c.img=45;

	printf("%d+%di",c.real,c.real);

	//array

	struct complex c1[10];

	//
	int n;
	printf("Enter n:");
	scanf("%d",&n);

	printf("Enter the values:\n\n");
	for(int i=0;i<n;i++){
		printf("real:");
		scanf("%d",&c1[i].real);
		printf("img:");
```

```
		scanf("%d",&c1[i].img);
		printf("\n\n");
	}

	printf("complex numbers:\n\n");

	for(int i=0;i<n;i++){
		printf("%d+%di\n\n",c1[i].real,c1[i].img);
	}
}

//
//3. Write a program to create an array for 10 players. For each player store name, no. of
//matches played, runs, wickets takes.
//a. Create function to Accept the information of each player.
//b. Create function to display the information of all the players
//c. Display the information of player who made maximum runs and the one who took
//maximum number of wickets.



#include<stdio.h>
typedef struct player {
	char name[20];
	int no_of_m;
	int runs;
	int wickets;
}player;

void store(player*,int);
void display(player*,int);
void max_run(player*,int);
```

```
void max_wicket(player*,int);



void main(){


        player p[10];
        int n;
        printf("enter the n:");
        scanf("%d",&n);
        store(p,n);
        display(p,n);
        max_run(p,n);
        max_wicket(p,n);
}
void store(player* p,int n){
        for(int i=0;i<n;i++){
        printf("Enter the details\n");
        fflush(stdin);
        printf("Name:");
        gets(p[i].name);
        printf("No_of_matches:");
        scanf("%d",&p[i].no_of_m);
        printf("Runs:");
        scanf("%d",&p[i].runs);
        printf("Wickets:");
        scanf("%d",&p[i].wickets);
        printf("\n\n");
        }



}
```

```
void display(player* p,int n){
	for(int i=0;i<n;i++){
		printf("playes Details:\n\n");
		printf("name:%s\n",p[i].name);
		//puts(p[i].name);
		printf("No_of_matches:%d\n",p[i].no_of_m);
		printf("no_of_runs:%d\n",p[i].runs);
		printf("NO_of_wickets:%d\n",p[i].wickets);
			printf("\n\n");
	}

}

void max_run(player* p,int n){

	int max=p[0].runs;
	for(int i=0;i<n;i++){
		if(p[i].runs>max){
			max=p[i].runs;
		}
	}
	printf("\n\n");
	printf("Max runs:%d",max);
}


void max_wicket(player* p,int n){
	int max=p[0].wickets;
	for(int i=0;i<n;i++){
		if(p[i].wickets>max){
```

```
                        max=p[i].wickets;
                }
        }
                printf("\n\n");
        printf("Max_wickets:%d",max);

}

//4. Point of Sale System: Build a simple point of sale system using structures to
//represent products with attributes like name, price, and quantity. Allow users
//to add items to a cart and calculate the total cost.

#include<stdio.h>
#include<string.h>
typedef struct product{
        char name[20];
        double price;
        int quantity;

}product;

int cnt_cart=0;

void display(product*,int);
void display_cart(product*,int);
void add_cart(product*,int,product*,int);

void main(){


        product p[5];
```

```
product cart[5];

int size=5;

//fill the array of product

strcpy(p[0].name,"shampoo");
p[0].price=45.3;
p[0].quantity=5;

strcpy(p[1].name,"biscuit");
p[1].price=50.3;
p[1].quantity=6;

strcpy(p[2].name,"soap");
p[2].price=50.3;
p[2].quantity=6;


strcpy(p[3].name,"book");
p[3].price=75.5;
p[3].quantity=10;

strcpy(p[4].name,"notebook");
p[4].price=75.5;
p[4].quantity=10;

int choice;

        do{
```

```
char nam[20];

printf("\n----------Welcome to the sales system----------\n\n");

printf("\n0.Exit\n1.Display Product\n2.Add Product\n");
printf("Enter the choice:");
scanf("%d",&choice);

switch(choice){
                                        case 0:
        {
                printf("\nThank you for visit!!\n");
                break;
        }
case 1:
        {

        display(p,size);
        break;
        }
case 2:
        {
                printf("Enter name:");
                scanf("%s",nam);
                                        int flag=0;
                                                for(int i=0;i<size;i++){
                                                        int
cmp=strcmp(p[i].name,nam);
                                                                                if(cmp==0){

        //ask for the quantity if not enough then display insufficient
```

```
                                                                    int
qnt;//that much quantity must reduce from total qnt

        printf("Enter the Quantity:\n");

        scanf("%d",&qnt);

        if(qnt<=p[i].quantity){

        add_cart(cart,i,p,qnt);

        flag=1;
                                                                    }


        else{

        printf("Insufficient Quantity!!");
                                                                    }

                                                                }

                                                        }

                                                            if(flag==0)
                                                                printf("product not
found!!");


                                                        display_cart(cart,size);


                                                        break;

                                    }
                            default:
```

```
                {
                        printf("Invalid Case!!");
                }
        }//switch


    }while(choice!=0);

}

void display(product* p,int size){
                printf("\nSales system\n");
                printf("+---------------+--------+-----------+\n");
                printf("|     NAME     |  Price | Quantity  |\n");
                printf("+---------------+--------+-----------+\n");
        for(int i=0;i<size;i++){

                printf("|%15s|%-8.2lf|%-11d|\n",p[i].name,p[i].price,p[i].quantity);
                printf("+---------------+--------+-----------+\n");
        }
    }
```

```
void add_cart(product* cart,int i,product* pro,int qnt){
                        //copy that entire element and then reduce quantity as per count
                        //here we dont want to copy as it is ,we want modify some values
                        strcpy(cart[cnt_cart].name,pro[i].name);
                        cart[cnt_cart].quantity=qnt;//as per quantity decide the price
                        cart[cnt_cart].price=qnt* pro[i].price;//modify the price
                        pro[i].quantity=pro[i].quantity-qnt;//reduce that much quantity from total
quantity
                        cnt_cart++;//maintain the size of cart array



                }

void display_cart(product* c,int size){
                        double total;
                        printf("\n----------YOUR CART----------\n");
                printf("+---------------+--------+-----------+\n");
                        printf("|     NAME     |  Price | Quantity  |\n");
                        printf("+---------------+--------+-----------+\n");
                for(int i=0;i<cnt_cart;i++){

                        printf("|%15s|%-8.2lf|%-11d|\n",c[i].name,c[i].price,c[i].quantity);
                        printf("+---------------+--------+-----------+\n");
                }
                for(int i=0;i<cnt_cart;i++){
                        total=total+c[i].price;
                }
        printf("|  total price  -->  RS. %-8.2lf\n\n",total);
        printf("Please pay the total amount to buy the product!!\n");
}
```

```
//5. Movie Database: Create a program that uses structures to manage a movie
//database with details like title, director, release year, and genre. Allow users
//to add, search for, and update movie records

#include<stdio.h>
#include<string.h>
typedef struct movie{
	char title[50];
	char director[50];
	int release_year;
	char genre[50];
}movie;

void display_movie(int,movie*);
void display_all(movie*,int);

void main(){
	movie miv[20];
	int n=0;
	int choice;
	do{
		printf("0.Exit\n1.add\n2.search\n3.update_records\n4.display\n");
		printf("Enter choice:");
		scanf("%d",&choice);

		switch(choice){
			case 0:{
				printf("Thanks for visit!!");
				break;
			}
			case 1:
```

```
{
        int num;
        printf("Enter the number of movies wants to enters:");
        scanf("%d",&num);
        for(int i=n;i<(n+num);i++){
                printf("Enter the movie details:\n");
                fflush(stdin);
                char tit[50];
                printf("Enter the title:");
                gets(tit);

                strcpy(miv[i].title,tit);

                fflush(stdin);

                char dir[50];
                printf("Enter the director:");
                gets(dir);
                strcpy(miv[i].director,dir);

                int rel_y;
                printf("Enter the release year:");
                scanf("%d",&rel_y);
                miv[i].release_year=rel_y;

                fflush(stdin);
                char grn[50];
                printf("Enter the genre:");
                gets(grn);
                strcpy(miv[i].genre,grn);
                printf("\n");
```

```
                }
                //update n==>
                n=n+num;

                break;
            }
            case 2:
                {
                    //search
                    fflush(stdin);
                    printf("Enter the title to search:");
                    char tit_m[50];
                    gets(tit_m);
                    int flag=0;
                    for(int i=0;i<n;i++){
                        int c=strcmp(miv[i].title,tit_m);
                        if(c==0){
                            flag=1;
//                          //display
//                          printf("Title:%s\n",miv[i].title);
//
        printf("Director:%s\n",miv[i].director);
//                          printf("Release Year
:%d\n",miv[i].release_year);
//                          printf("Genre:%s\n",miv[i].genre);
                            display_movie(i,miv);

                            break;//as got stop search
                        }
                    }//out of for loop

                    if(flag==0){
```

```
                                        printf("Moive not found!!");
                                }

                                        break;
                        }
                case 3:{
                        //update
                        fflush(stdin);
                        char t[50];
                        int flag=0;
                        printf("Enter the movie title wants to change:");
                        gets(t);
                        int i;
                        for(i=0;i<n;i++){
                                if(strcmp(miv[i].title,t)==0){
                                        flag=1;

        printf("1.title\n2.director\n3.Release Year\n4.genre\n");
                                                                                                int
c;

        scanf("%d",&c);

        switch(c){

        case 1:

                {       fflush(stdin);

                        char title[50];

                        printf("Enter the new title:");

                        gets(title);
```

```
                strcpy(miv[i].title,title);
                display_movie(i,miv);
                break;
            }
case 2:{
            fflush(stdin);
            char direct[50];
            printf("Enter the new director:");
            gets(direct);
            strcpy(miv[i].director,direct);
            display_movie(i,miv);
            break;
}
case 3:{
            int r_year;
            printf("Enter the new year:");
            scanf("%d",&r_year);
            miv[i].release_year=r_year;
            display_movie(i,miv);
```

```
                break;
        }
        case 4:{
                fflush(stdin);
                char g[50];
                printf("Enter the new genre:");
                gets(g);
                strcpy(miv[i].genre,g);
                display_movie(i,miv);
                break;
        }
        default:
                {
                        printf("Invalid Input!!");
                        break;
                }
                                                                        }
                                                                //break the
for loop
                                                                break;
```

```
                                        }//if in for
                                    }//end for

                                    //display updated movie


                                if(flag==0){
                                        printf("Movie not found!!");
                                }
                                        break;
                                }
                        case 4:{
                                 display_all(miv,n);
                                break;
                        }
                        default:{
                                printf("Invalid choice!!");
                                break;
                        }
                }
        }while(choice!=0);
}

void display_movie(int i,movie*miv){

                                                        printf("Title:%s\n",miv[i].title);

        printf("Director:%s\n",miv[i].director);
                                                        printf("Release Year
:%d\n",miv[i].release_year);
                                                        printf("Genre:%s\n",miv[i].genre);
        }
```

```
void display_all(movie* miv,int n){
	for(int i=0;i<n;i++){
		printf("Title:%s\n",miv[i].title);
	printf("Director:%s\n",miv[i].director);
		printf("Release Year
:%d\n",miv[i].release_year);
		printf("Genre:%s\n",miv[i].genre);
		printf("\n");
	}
}
```

//2. Create a structure Time with data members as hrs, min, sec. Accept the values of all
//these members from user and display them. Also perform addition of two time variables
//and display the result. If sec goes beyond 60, carry it to min etc. Add a method to convert
//the given time into sec.

```
#include<stdio.h>
typedef struct time{
	int hr;
	int min;
	int sec;
}time;

void display(time);
void display(time);
```

```
void main(){
	time t1;
	time t2;
	//t3 for addition
	time t3;

	store(&t1);
	store(&t2);

	display(t1);
	display(t2);

//	t3.hr=t1.hr+t2.hr;
//	t3.min=t1.hr+t2.min;
//	t3.sec=t1.sec+t2.sec;

	//display(t3);


}

void store(time* t){
	printf("Enter the details:\n");
	printf("Hr:");
	scanf("%d",&t->hr);
	printf("Min:");
	scanf("%d",&t->min);
	printf("Sec:");
	scanf("%d",&t->sec);
}
```

```
void display(time t){
	if(t.sec>59){
		int r = t.sec%60;
		int q = t.sec/60;//t.sec will modify after
		t.sec = r;
		t.min=t.min+(q);
//	printf("t.sec:%d\n",t.sec);
//	printf("t.min:%d\n",t.min);
	}
	if(t.min>59){
		int r=t.min%60;
		int q=t.min/60;
		t.min=r;
		t.hr=t.hr+(q);
//		printf("t.min:%d\n",t.min);
//		printf("t.hr:%d\n",t.hr);
	}
	printf("%d:%d:%d\n\n",t.hr,t.min,t.sec);

}
```