DS Assignment-3

---

1. In a classroom of 100 students you have to divide students in 10 sub list of equal

students.

Perfrom the following operation.

1. Insertation of marks.

2. Deletion of marks.

3. Updation of marks.

4. Display marks

5. Replace one sub list with other

6. Compare two sub list

7. Enter nth element in jth list

Code:

```
#include<stdio.h>

#include <stdlib.h>

int students[50]={};


int rollNo(int sublist,int lastdigit){

        return (sublist*10)+lastdigit-1;
```

```
}
void insertion(int marks,int sublist1, int lastdigit){
	students[rollNo(sublist1,lastdigit)]=marks;
	printf("Your marks have been inserted.\n");
}
void deletion(int sublist1, int lastdigit){
	int roll=rollNo(sublist1,lastdigit);
	students[rollNo(sublist1,lastdigit)]=0;
	printf("Your marks have been deleted.\n");
}
void updation(int marks,int sublist1, int lastdigit){
	int roll=rollNo(sublist1,lastdigit);
	students[roll]=marks;
	printf("Your marks have been updated.\n");
}
void display(int sublist1, int lastdigit){
	int roll=rollNo(sublist1,lastdigit);
	printf("Your marks are: %d\n",students[roll]);
}
void replace(int sublist1,int sublist2){
	int temp,i;
	for(i=1;i<=10;i++){
```

```
            temp=students[rollNo(sublist1,i)];

        students[rollNo(sublist1,i)]=students[rollNo(sublist2,i)];
            students[rollNo(sublist2,i)]=temp;
        }
        printf("The sublists have been replaced\n");
}
void compare(int sublist1,int sublist2){
        int i,count=0;
        for(i=1;i<=10;i++){

        if(students[rollNo(sublist1,i)]==students[rollNo(sublist2,i
)])
                {
                        count++;
                        continue;
                }else
                        break;
        }
        if(count==10){
                printf("The sublists are equal\n");
        }else{
                printf("The sublists are not equal\n");
```

```
        }
}
void sublist(int sublist){
        int i;
        printf("Enter the sublist values between 0 and 100: ");
        for(i=1;i<=10;i++){
                scanf("%d",&students[rollNo(sublist,i)]);
        }
}

int main(){

        printf("Enter your
choice:\n1.Insertion\n2.Deletion\n3.Updation\n4.Display
marks\n");
        printf("5.Replace\n6.Compare\n7.Enter nth element in
jth list\n8.Insert Sublist\n9.Exit\n");
        while(1){
        printf("\nEnter your choice: ");
        int ch,a,b,marks;
        scanf("%d",&ch);
        switch(ch){
                case 1:
```

```
                printf("Enter the sub list and element and
marks: ");
                scanf("%d %d %d",&a,&b,&marks);
                if((marks<0||marks>100)){
                        printf("Please enter valid marks\n");
                        break;
                }
                if(a<0 || a>4|| b<0 || b>9){
                        printf("Please give valid input\n");
                        break;
                }
                insertion(marks,a,b);
                break;
            case 2:
                printf("Enter the sub list and element: ");
                scanf("%d %d",&a,&b);
                if(a<0 || a>4|| b<0 || b>9){
                        printf("Please give valid input\n");
                        break;
                }
                deletion(a,b);
                break;
```

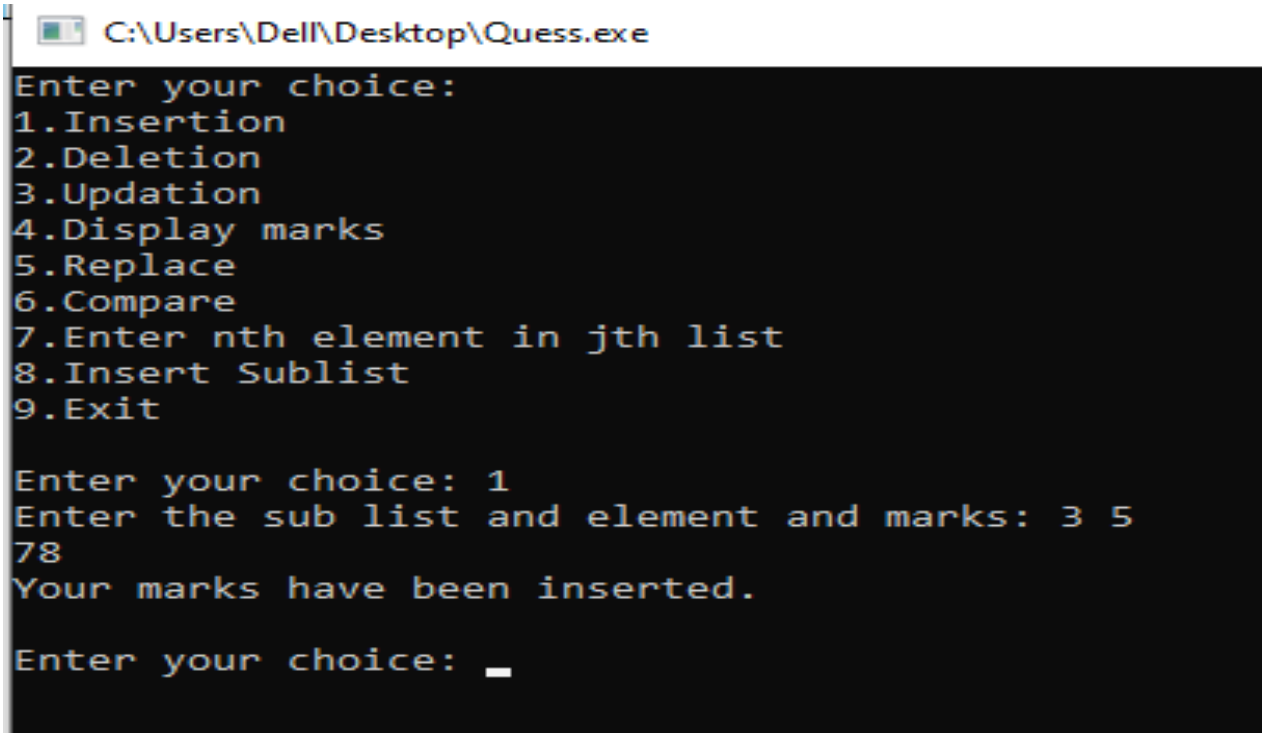
```
            case 3:
                printf("Enter the sublist and the element and
new marks: ");
                scanf("%d %d %d",&a,&b,&marks);
                if(a<0 || a>4|| b<0 || b>9){
                    printf("Please give valid input\n");
                    break;
                }
                if((marks<0||marks>100)){
                    printf("Please enter valid marks\n");
                    break;
                }
                updation(marks,a,b);
                break;
            case 4:
                printf("Enter the sublist and the element: ");
                scanf("%d %d",&a,&b);
                if(a<0 || a>4|| b<0 || b>9){
                    printf("Please give valid input\n");
                    break;
                }
                display(a,b);
```

```
        break;
case 5:
        printf("Enter the sublist 1 and 2: ");
        scanf("%d %d",&a,&b);
        if(a<0 || a>4|| b<0 || b>4){
                printf("Please give valid input\n");
                break;
        }
        replace(a,b);
        break;
case 6:
        printf("Enter the sublist 1 and 2: ");
        scanf("%d %d",&a,&b);
        if(a<0 || a>4|| b<0 || b>4){
                printf("Please give valid input\n");
                break;
        }
        compare(a,b);
        break;
case 7:
        printf("Enter the nth element and jth list: ");
        scanf("%d %d",&a,&b);
```

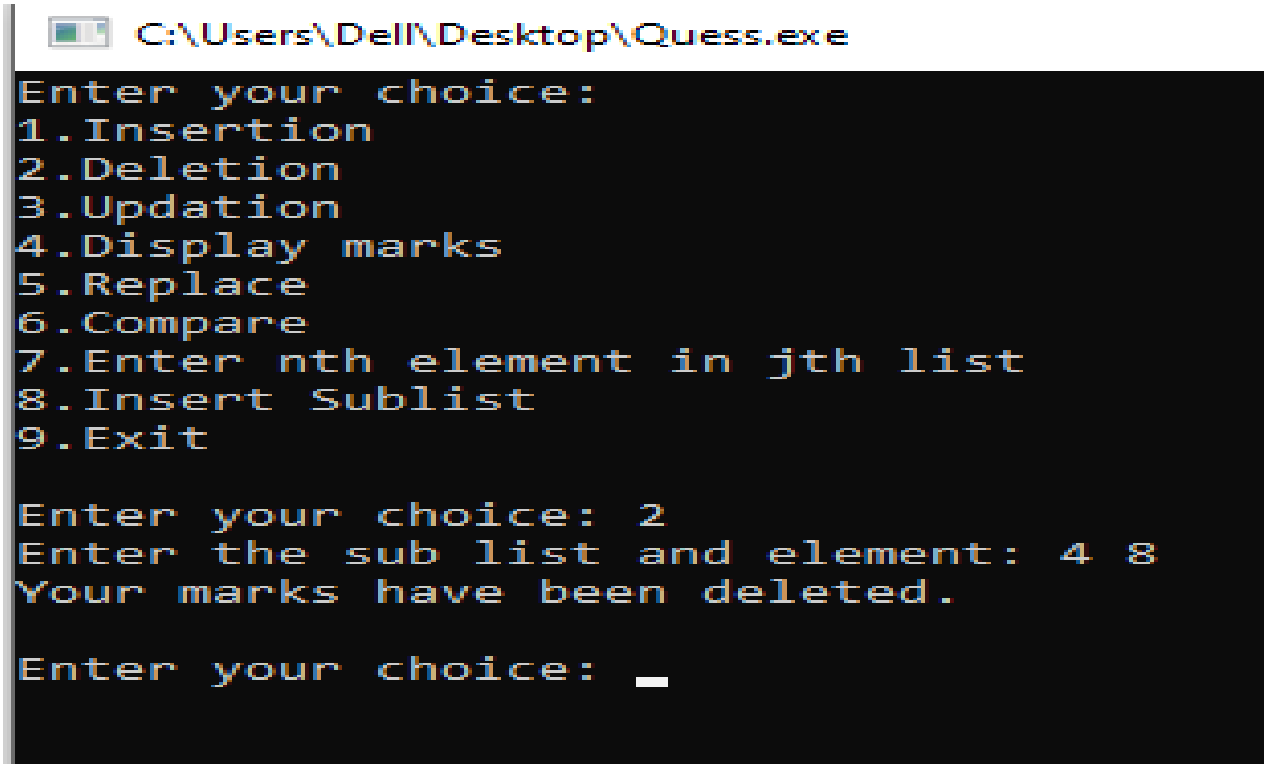
```
                if(a<0 || a>4|| b<0 || b>9){
                        printf("Please give valid input\n");
                        break;
                }
                printf("The roll no is: %d\n",rollNo(a,b)+1);
                break;
        case 8:
                printf("Enter the nth element (sublist): ");
                scanf("%d",&a);
                sublist(a);
                break;
        case 9:
                exit(0);
        default:
                printf("Wrong Choice\n");
    }
  }
}
```

Output:

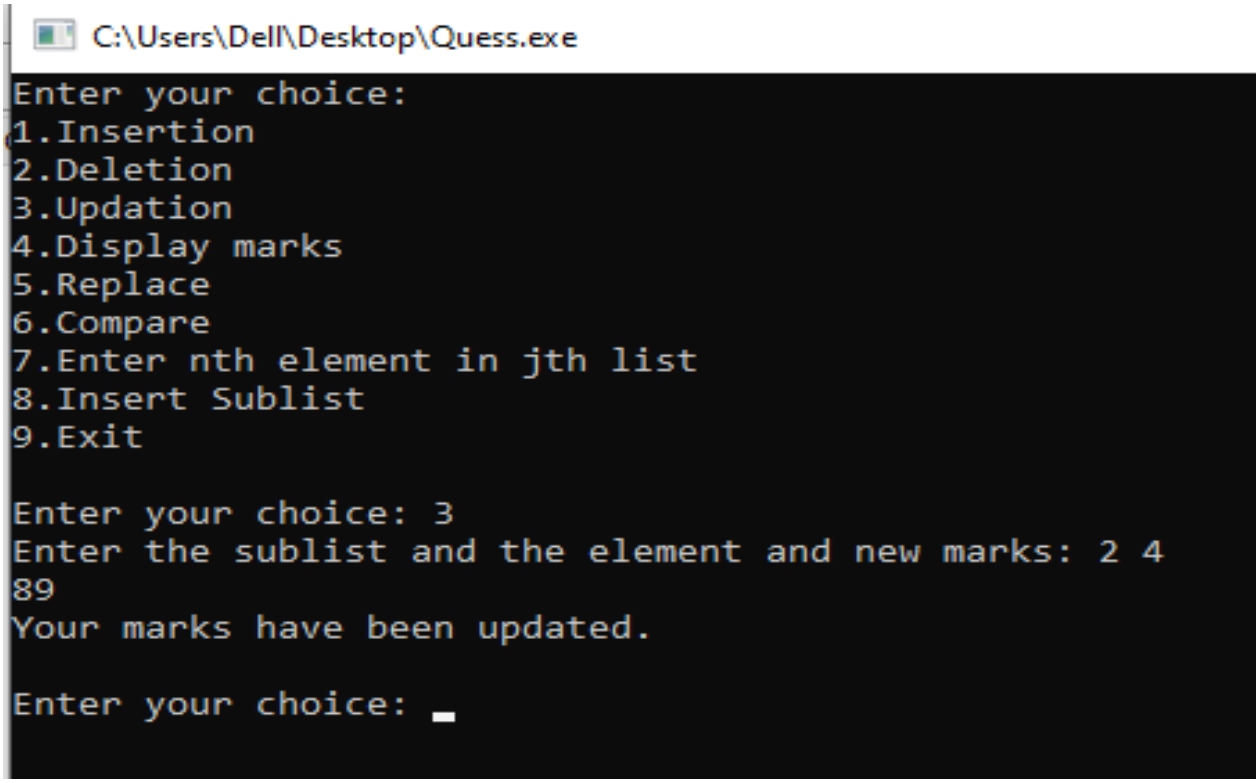
1) Insertion



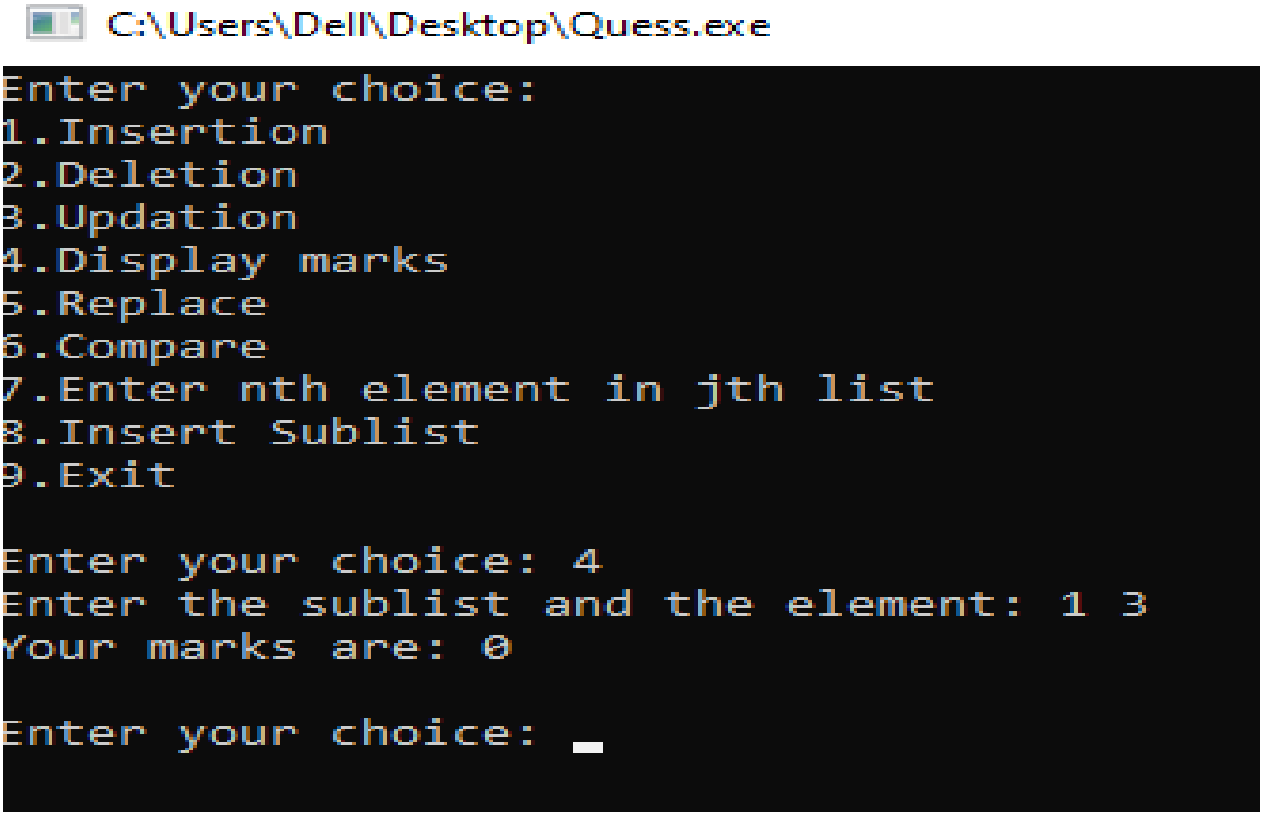
2) Deletion



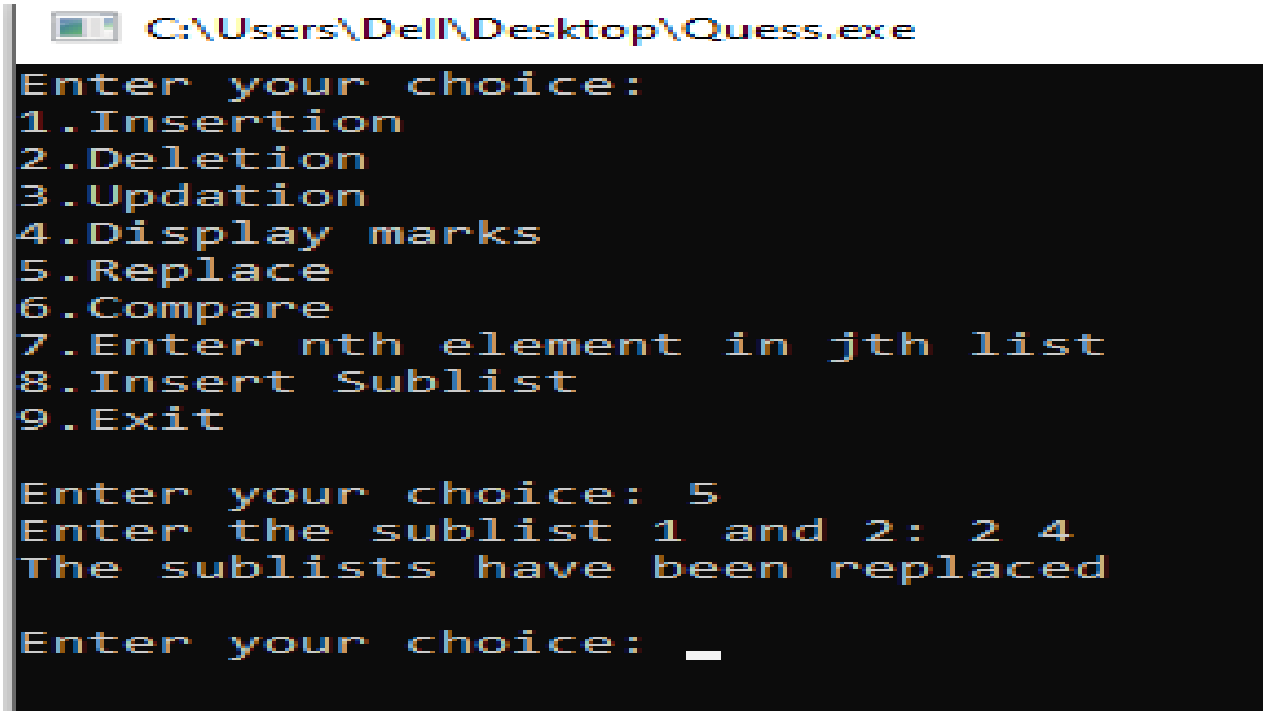
3) Updation



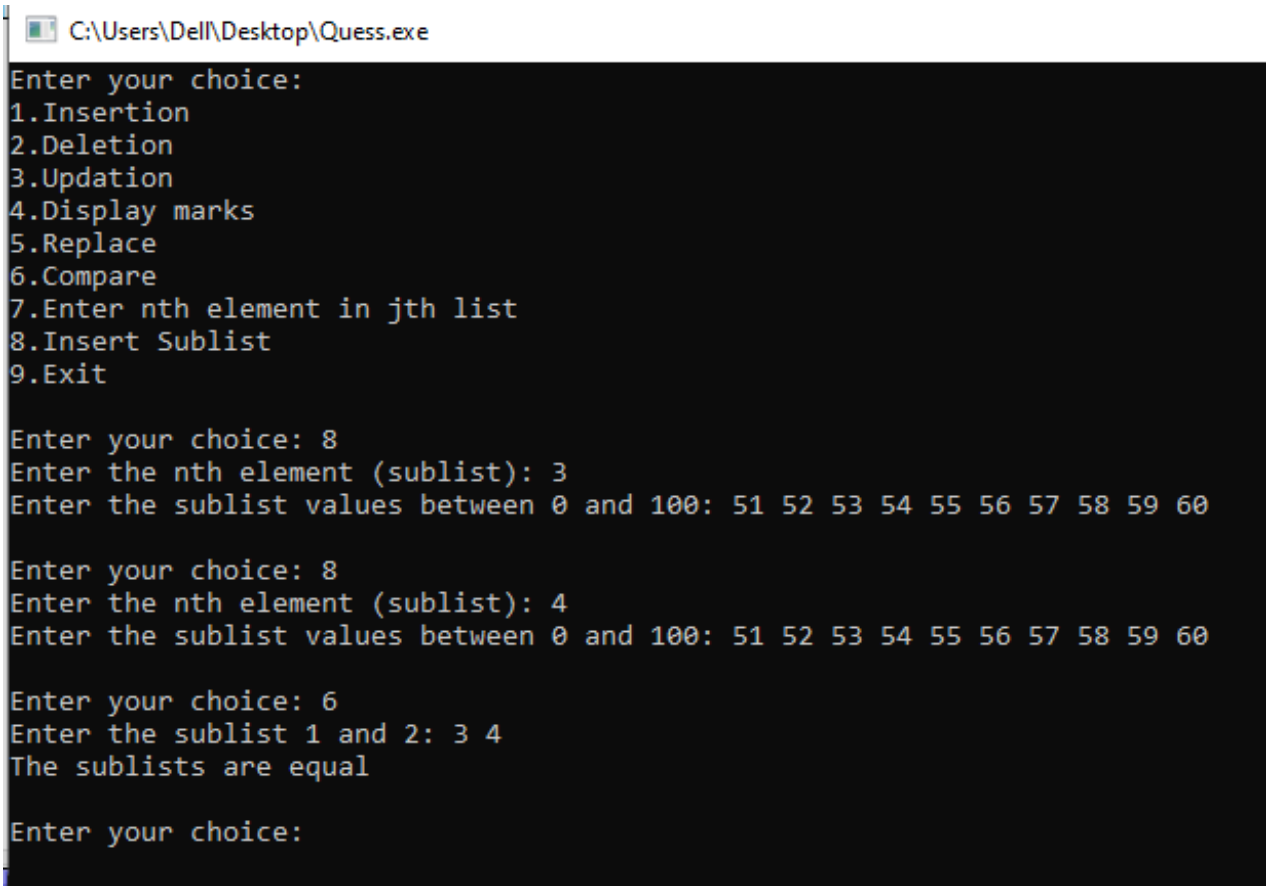
4) Display Marks



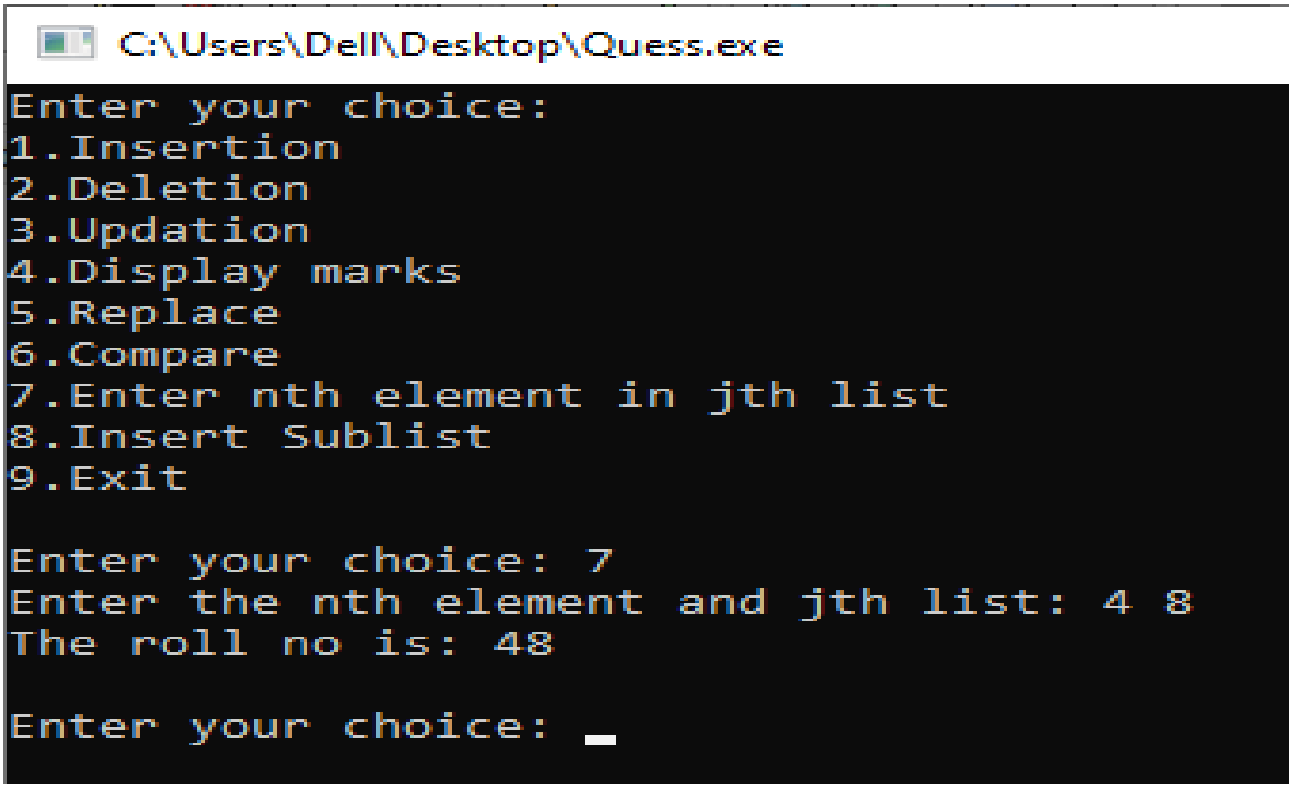
5) Replace



6) Compare



7) Enter nth element in jth list



2) Write a program to generate sub matrices from the multidimensional matrix.

Take any two sub matrices and perform the addition of two sub matrices.

For example input Array is A[4][4] and B[2][2]

Find the occurrence of B in A. The rows or columns in A are not required to be

consecutive.

Code:

```
#include<stdio.h>

int main()
```

```
{
	int mat[3][3];
	int i=0,j=0;
	printf("enter the elements of array: \n");
	for(i=0;i<3;i++)
	{
		for(j=0;j<3;j++)
		{
			scanf("%d",&mat[i][j]);
			printf(" ");
		}
		printf("\n");
	}
	//[1][1] sub matrix
	printf("[1][1] sub matrix:\n");
	for(i=0;i<3;i++)
	{
		for(j=0;j<3;j++)
		{
			printf("[%d] \n",mat[i][j]);
```

```
        }

}
printf("[2][2] sub matrix:\n");
int l=0,k=0;
for(l=0;l<2;l++)
{
        for(k=0;k<2;k++)
        {
                for(i=l;i<l+2;i++)
                {
                        for(j=k;j<k+2;j++)
                        {
                                printf("%d ",mat[i][j]);
                        }
                        printf("\n");

                }
                printf("\n ---------\n");
        }
```

```
    }
    printf("addition of [2][2] matrix1+matrix2: \n");
    int mat1[2][2];
    for(i=0;i<2;i++)
    {
        for(j=0;j<2;j++)
        {
            mat1[i][j]=mat[i][j]+mat[i][j+1];
        }
    }
    for(i=0;i<2;i++)
    {
        for(j=0;j<2;j++)
        {
            printf("%d ",mat1[i][j]);
        }
        printf("\n");
    }
}
```

Output:

C:\Users\Dell\Downloads\q2 (2).exe

[redacted]

[redacted]

[redacted]

```
5  6

-----------
4  5
7  8

-----------
5  6
8  9
```

```
----------
addition of [2][2] matrix1+matrix2:
3 5
9 11

--------------------------------
Process exited after 5.697 seconds with return value 10
Press any key to continue . . .
```