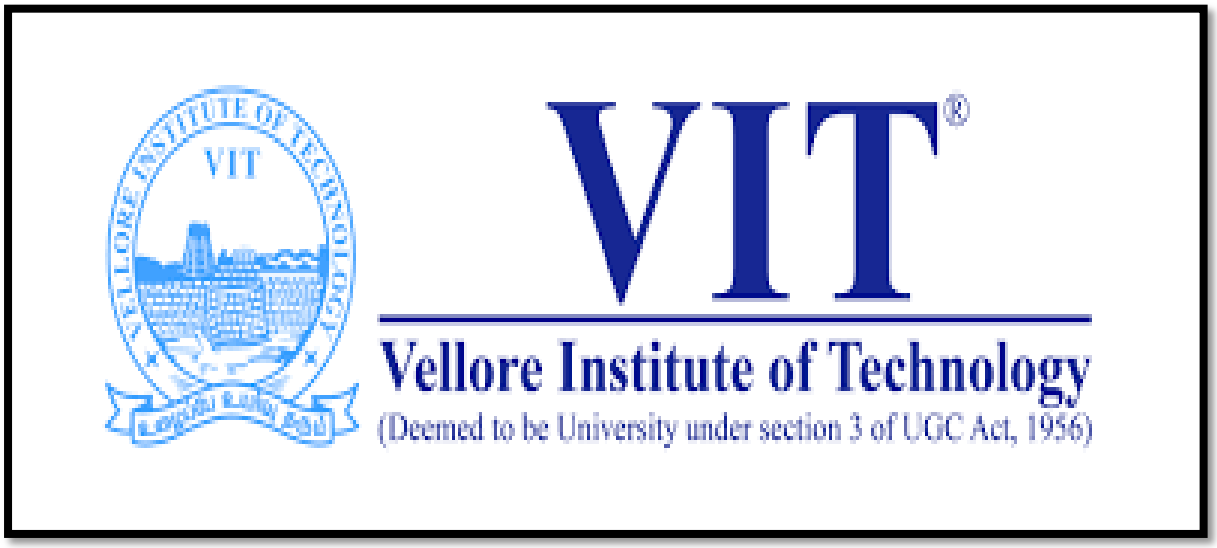


NAME : NITHISH.G

REG NO : 19BCS0012

SUBJECT : DATA STRUCTURE

COURSE : CSC2001

CODE

SLOT : ETH+TE1

1. **Using structures concept write a program to read and display the information of all the students in a class. Then edit the details of the ith student and redisplay the entire information**

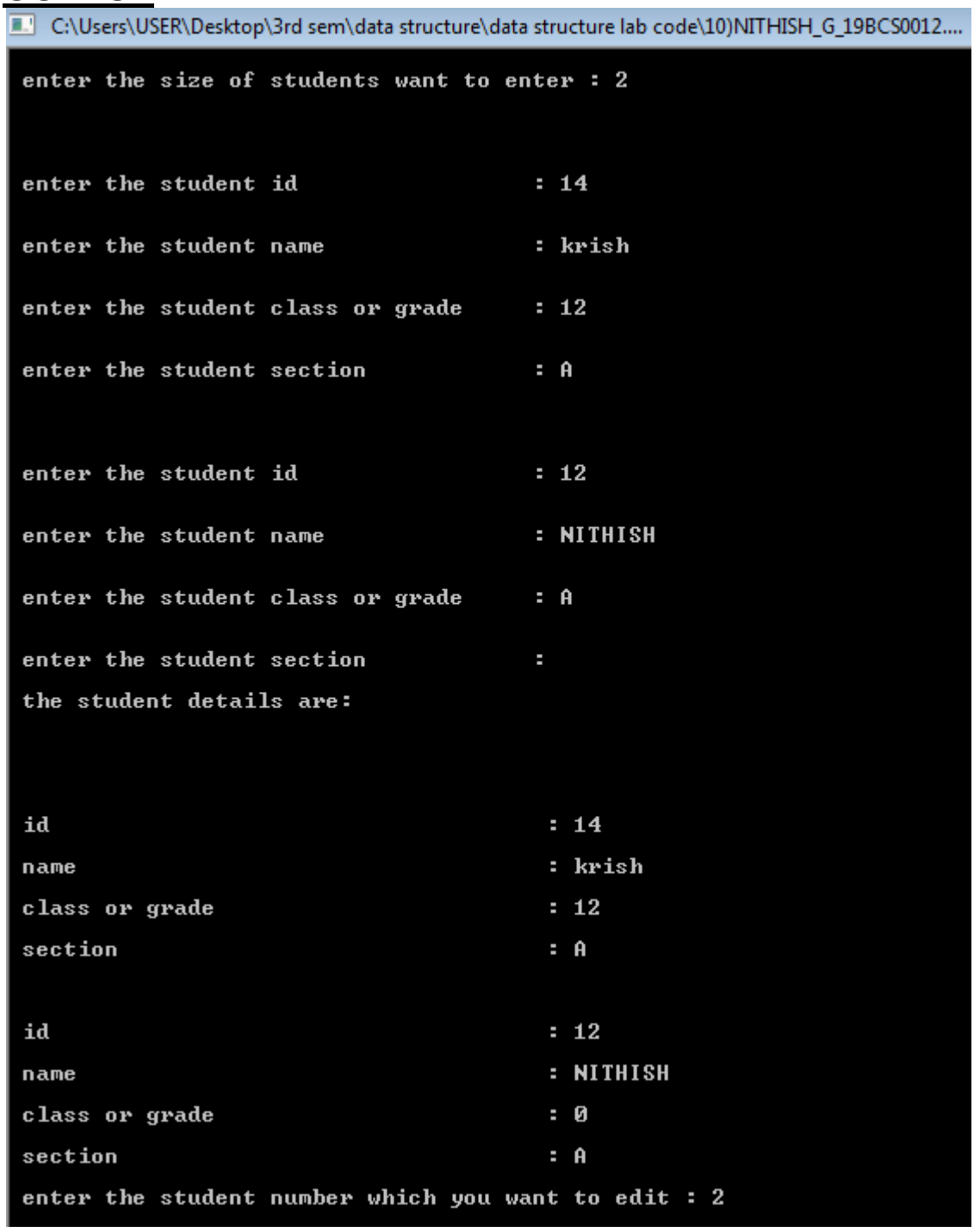
```
#include<conio.h>
#include<stdio.h>

struct std
{
   int id;
   char name[100];
   int cls;
   char section[1];
}s[100];

int main()
{
   int n,i,e;
printf("\n enter the size of students want to enter : ");
scanf("%d",&n);
   for(i=1;i<=n;i++)
   {
printf("\n\n");
printf("\n\n enter the student id : ");
scanf("%d",&s[i].id);
printf("\n\n enter the student name : ");
scanf("%s",s[i].name);
printf("\n\n enter the student class or grade : ");
```

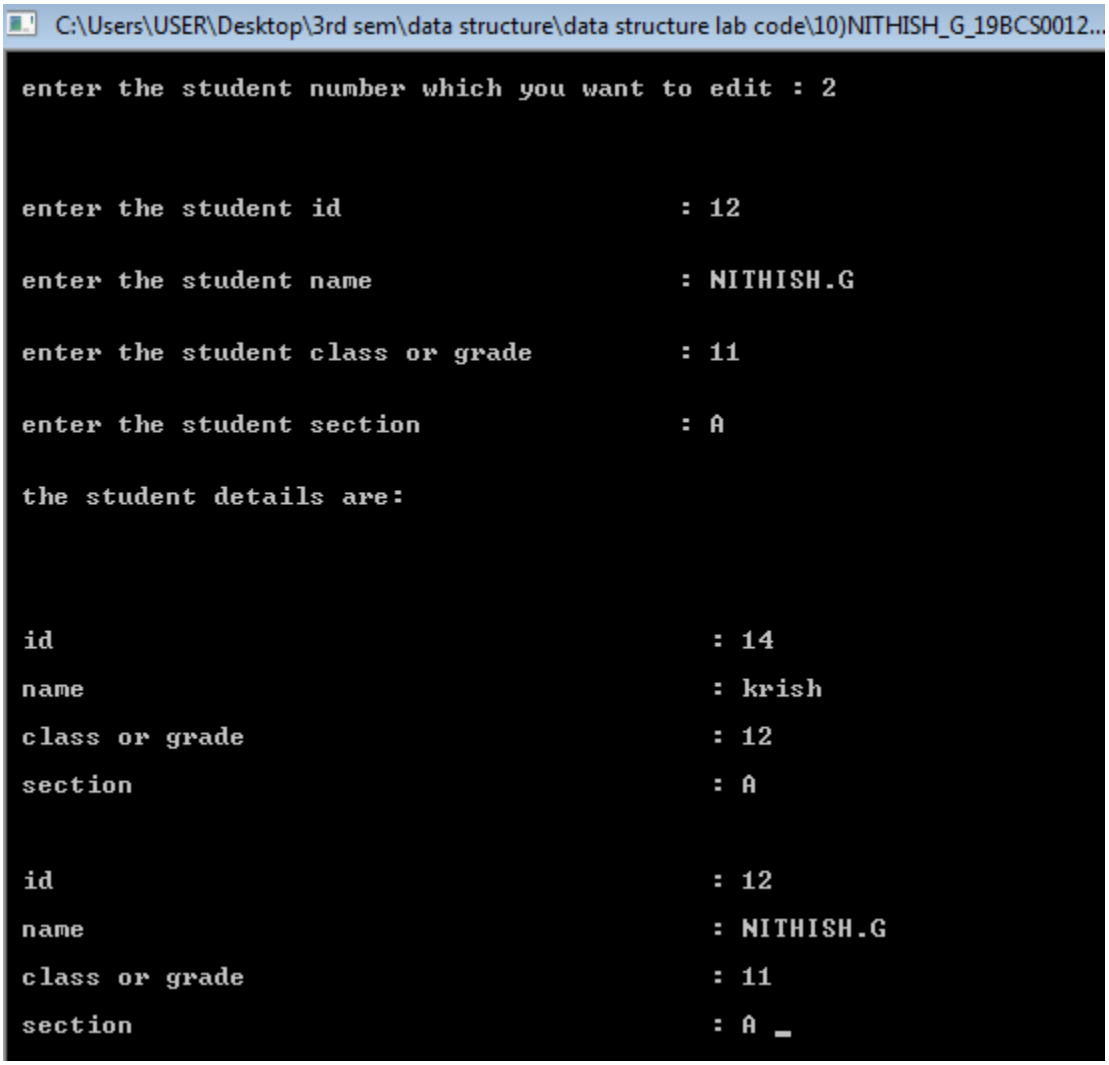
```
scanf("%d",&s[i].cls);
printf("\n\n enter the student section : ");
scanf("%s",s[i].section);
   }
printf("\n\n the student details are:\n\n");
   for(i=1;i<=n;i++)
   {
printf("\n\n");
printf("\n\n id : %d ",s[i].id);
printf("\n\n name : %s ",s[i].name);
printf("\n\n class or grade : %d ",s[i].cls);
printf("\n\n section : %s ",s[i].section);

   }
printf("\n\n enter the student number which you want to edit : ");
scanf("%d",&e);
   int flag = 0;
   for(i=1;i<=n;i++)
   {
   if(i==e)
   {
      flag=1;
printf("\n\n");
printf("\n\n enter the student id : ");
scanf("%d",&s[e].id);
printf("\n\n enter the student name : ");
scanf("%s",s[e].name);
printf("\n\n enter the student class or grade : ");
scanf("%d",&s[e].cls);
```

```
printf("\n\n enter the student section : ");
scanf("%s",s[e].section);
    }
    }
   if(flag==0)
   {
printf("\n\n enter the proper student number between 1 to %d : ",n);
scanf("%d",&e);

        for(i=1;i<=n;i++)
        {
            if(i==e)
            {
                flag=1;
printf("\n\n");
printf("\n\n enter the student id : ");
scanf("%d",&s[e].id);
printf("\n\n enter the student name : ");
scanf("%s",s[e].name);
printf("\n\n enter the student class or grade : ");
scanf("%d",&s[e].cls);
printf("\n\n enter the student section : ");
scanf("%s",s[e].section);
            }
        }
   }
printf("\n\n the student details are:\n\n");
    for(i=1;i<=n;i++)
    {
```

```
printf("\n\n");
printf("\n\n id : %d ",s[i].id);
printf("\n\n name : %s ",s[i].name);
printf("\n\n class or grade : %d ",s[i].cls);
printf("\n\n section : %s ",s[i].section);

    }
    getch();
    return 0;
}
```

OUTPUT


C:\Users\USER\Desktop\3rd sem\data structure\data structure lab code\10)NITHISH_G_19BCS0012....
enter the size of students want to enter : 2
enter the student id : 14
enter the student name : krish
enter the student class or grade : 12
enter the student section : A
enter the student id : 12
enter the student name : NITHISH
enter the student class or grade : A
enter the student section :
the student details are:
id : 14
name : krish
class or grade : 12
section : A
id : 12
name : NITHISH
class or grade : 0
section : A
enter the student number which you want to edit : 2



2. Explain the drawbacks of pointers

Drawbacks of pointers in c:

- **Uninitialized pointers might cause segmentation fault.**
- **Dynamically allocated block needs to be freed explicitly. Otherwise, it would lead to memory leak.**
- **Pointers are slower than normal variables.**

- If pointers are updated with incorrect values, it might lead to memory corruption.

- Basically, pointer bugs are difficult to debug. Its programmer's responsibility to use pointers effectively and correctly.

3. Explain with an example how the structures members can be passed through pointers.

Access members using Pointer

To access members of a structure using pointers, we use the -> operator.

- Method of accessing members of the structure using pointers is slightly confusing and less readable, that's why C provides another way to access members using the arrow (`->`) operator. To access members using arrow (`->`) operator write pointer variable followed by `->` operator, followed by name of the member.

SOURCE CODE

```
#include <stdio.h>
#include<conio.h>
#include <string.h>
struct student
{
    int id;
```

```
    char name[30];
    float percentage;
};
int main()
{
    int i;
    struct student record1 = {1, "Raju", 90.5};
    struct student *ptr;

    ptr = &record1;

       printf("Records of STUDENT1: \n");
       printf("  Id is: %d \n", ptr->id);
       printf("  Name is: %s \n", ptr->name);
       printf("  Percentage is: %f \n\n", ptr->percentage);
getch();
    return 0; }
```

OUT PUT