Computer Engineering Department, S.V.N.I.T. Surat.

B Tech (CO) –IInd Year semester-III

Course: *Data Structures CO203*

**Assignment-IV**

1.) Create an array to store the student information (ex. Student code, Name, Address, Mobile, Total mark, CGPA)

Perform following operations:

(1) Insert

(2) Delete

(3) Search

(4) Traverse

(5) Update the element at particular position given by the user.

Code:

*// Create an array to store the student information (ex. Student code, Name, Address, Mobile, Total mark, CGPA)*

*// Perform following operations:*

*// (1) Insert*

*// (2) Delete*

*// (3) Search*

*// (4) Traverse*

*// (5) Update the element at particular position given by the user.*

*#include* <stdio.h>

*#include* <string.h>

*// Declaring a structure class*

struct student

{

    int s\_code;

    char s\_name[100];

    char s\_address[10000];

    int s\_mark;

    float s\_cgpa;

};

struct student stud[101]; *// Array of Student Structure*

int i, j;

int idx = 0; *// Takes Care of Number of Element Inserted*

*// Common Function for Printing Data*

void print\_data(int id)

{

    printf("\nStudent Data\n");

    printf("Student Code    : %d\n", stud[id].s\_code);

    printf("Student Name    : %s", stud[id].s\_name);

    printf("Student Address : %s", stud[id].s\_address);

    printf("Student Mark    : %d\n", stud[id].s\_mark);

    printf("Student CGPA    : %f\n", stud[id].s\_cgpa);

}

*// 1. Insertation of Student Data*

void fx1()

{

    int cnt = 0;

    printf("Enter the Number of Students whose Data is to be Inserted :\n");

    scanf("%d", &cnt);

    fflush(stdin); *// for another input*

    int choice;

    printf("Would you Like to Insert by Position(1 -> yes && 0 -> no)?");

    scanf("%d", &choice);

*// INSERTION LOGIC ADDED*

*// BASED ON USER CHOICE*

*if* (!choice)

    {

*for* (int i = 1; i <= cnt; i++)

        {

*if* (idx == 100)

            {

                printf("No More Student Data can be Entered!\n");

            }

*else*

            {

                printf("Enter Student %d Details :\n", i);

                printf("Student Code :\n");

                scanf("%d", &stud[idx].s\_code);

                fflush(stdin);

                printf("Student Name :\n");

                fgets(stud[idx].s\_name, sizeof(stud[idx].s\_name), stdin);

                printf("Student Address :\n");

                fflush(stdin);

                fgets(stud[idx].s\_address, sizeof(stud[idx].s\_address), stdin);

                fflush(stdin);

                printf("Student Mark [1-100] :\n");

                scanf("%d", &stud[idx].s\_mark);

                printf("Student CGPA [1-10] :\n");

                scanf("%f", &stud[idx].s\_cgpa);

                idx += 1;

            }

        }

    }

*else*

    {

*for* (int i = 1; i <= cnt; i++)

        {

*if* (idx == 100)

            {

                printf("No More Student Data can be Entered!\n");

            }

*else*

            {

                int pos;

                printf("Enter the Position where Student %d Data Needs to Be Inserted :\n", i);

                scanf("%d", &pos);

*if* (pos - 1 >= 0 && pos - 1 <= idx)

                {

*// Shift all Elements(right of pos) to right*

*for* (int k = idx + 1; k >= pos; k--)

                    {

                        stud[k].s\_code = stud[k - 1].s\_code;

                        strcpy(stud[k].s\_name, stud[k - 1].s\_name);

                        strcpy(stud[k].s\_address, stud[k - 1].s\_address);

                        stud[k].s\_mark = stud[k - 1].s\_mark;

                        stud[k].s\_cgpa = stud[k - 1].s\_cgpa;

                    }

*// space made for new data*

                    printf("Enter Student %d Details :\n", i);

                    printf("Student Code :\n");

                    scanf("%d", &stud[pos - 1].s\_code);

                    fflush(stdin);

                    printf("Student Name :\n");

                    fgets(stud[pos - 1].s\_name, sizeof(stud[pos - 1].s\_name), stdin);

                    printf("Student Address :\n");

                    fflush(stdin);

                    fgets(stud[pos - 1].s\_address, sizeof(stud[pos - 1].s\_address), stdin);

                    fflush(stdin);

                    printf("Student Mark [1-100] :\n");

                    scanf("%d", &stud[pos - 1].s\_mark);

                    printf("Student CGPA [1-10] :\n");

                    scanf("%f", &stud[pos - 1].s\_cgpa);

                    idx += 1;

                }

*else*

                {

                    printf("Enter Valid Position!\n");

                }

            }

        }

    }

*return*;

}

*// 2. Deletion of Student Data*

void fx2()

{

    int cnt = 0;

    printf("Enter the Number of Students whose Data is to be Deleted :\n");

    scanf("%d", &cnt);

*for* (int i = 0; i < cnt; i++)

    {

        int rno;

        printf("Enter the Roll Number of Student :\n");

        scanf("%d", &rno);

        int found = 0;

*for* (j = 0; j < idx; j++)

        {

*if* (stud[j].s\_code == rno)

            {

                stud[j].s\_code = -1;

*// SHIFTING LOGIC ADDED*

*for* (int k = idx - 2; k >= j; k--)

                {

                    stud[k].s\_code = stud[k + 1].s\_code;

                    strcpy(stud[k].s\_name, stud[k + 1].s\_name);

                    strcpy(stud[k].s\_address, stud[k + 1].s\_address);

                    stud[k].s\_mark = stud[k + 1].s\_mark;

                    stud[k].s\_cgpa = stud[k + 1].s\_cgpa;

                }

                stud[idx - 1].s\_code = -1;

                idx -= 1; *// Decrease Size by 1 Element*

                printf("Student Data Deleted Succesfully!\n");

                found = 1;

*break*;

            }

        }

*if* (found == 0)

        {

            printf("Student Data Does not Exist!\n");

            printf("Either Data not Inserted by Admin or Deleted!\n");

        }

    }

*return*;

}

*//3. Search of Student Data*

void fx3()

{

    int cnt = 0;

    printf("Enter the Number of Students whose Data is to be Searched :\n");

    scanf("%d", &cnt);

*for* (int i = 0; i < cnt; i++)

    {

        int rno;

        printf("Enter the Roll Number of Student :\n");

        scanf("%d", &rno);

        int found = 0;

*for* (int j = 0; j < idx; j++)

        {

*if* (stud[j].s\_code == rno)

            {

                printf("Student Found!\n");

                print\_data(j);

                found = 1;

*break*;

            }

        }

*if* (found == 0)

        {

            printf("Student Data Does not Exist!\n");

            printf("Either Data not Inserted by Admin or Deleted!\n");

        }

    }

*return*;

}

*//4.Traverse of Student Data*

void fx4()

{

*for* (int k = 0; k < idx; k++)

    {

*if* (stud[k].s\_code != -1)

        {

            print\_data(k);

        }

    }

*return*;

}

*//5. Update the element at particular Roll Number given by the user.*

void fx5()

{

    int cnt = 0;

    printf("Enter the Number of Students whose Data is to be Updated :\n");

    scanf("%d", &cnt);

*for* (int i = 0; i < cnt; i++)

    {

        int rno;

        printf("Enter the Roll Number of Student :\n");

        scanf("%d", &rno);

        int found = 0;

*for* (int j = 0; j < idx; j++)

        {

*if* (stud[j].s\_code == rno)

            {

                printf("Enter Student Details :\n", i);

                printf("Student Code :\n");

                scanf("%d", &stud[j].s\_code);

                fflush(stdin);

                printf("Student Name :\n");

                fgets(stud[j].s\_name, sizeof(stud[j].s\_name), stdin);

                printf("Student Address :\n");

                fflush(stdin);

                fgets(stud[j].s\_address, sizeof(stud[j].s\_address), stdin);

                fflush(stdin);

                printf("Student Mark [1-100] :\n");

                scanf("%d", &stud[j].s\_mark);

                printf("Student CGPA [1-10] :\n");

                scanf("%f", &stud[j].s\_cgpa);

                found = 1;

*break*;

            }

        }

*if* (found == 0)

        {

            printf("Student Data Does not Exist in DataBase!\n");

            printf("Enter Student Details Again :\n");

            printf("Student Code :\n");

            scanf("%d", &stud[idx].s\_code);

            fflush(stdin);

            printf("Student Name :\n");

            fgets(stud[idx].s\_name, sizeof(stud[idx].s\_name), stdin);

            printf("Student Address :\n");

            fflush(stdin);

            fgets(stud[idx].s\_address, sizeof(stud[idx].s\_address), stdin);

            fflush(stdin);

            printf("Student Mark [1-100] :\n");

            scanf("%d", &stud[idx].s\_mark);

            printf("Student CGPA [1-10] :\n");

            scanf("%f", &stud[idx].s\_cgpa);

            idx += 1;

        }

    }

*return*;

*return*;

}

int main()

{

*// Marks of Student Not Initialized = -1*

*for* (int i = 0; i <= 100; i++)

    {

        stud[i].s\_code = -1;

    }

    int choice = 0;

    printf("Perform following operations:\n");

    printf("1 -> Insert\n");

    printf("2 -> Delete\n");

    printf("3 -> Search\n");

    printf("4 -> Traverse\n");

    printf("5 -> Update the element at particular Roll Number\n");

    char ch = 'Y';

*while* (ch == 'Y' || ch == 'y')

    {

        int choice = 0;

        printf("Enter your Choice : ");

        scanf("%d", &choice);

*switch* (choice)

        {

*case* 1:

            fx1();

*break*;

*case* 2:

            fx2();

*break*;

*case* 3:

            fx3();

*break*;

*case* 4:

            fx4();

*break*;

*case* 5:

            fx5();

*break*;

*default*:

            printf("Enter Valid Choice [1-5] Only!");

*break*;

        }

        fflush(stdin); *// for character input*

        printf("Want to Do Another Operation?(Y/N)\n");

        scanf("%c", &ch);

    }

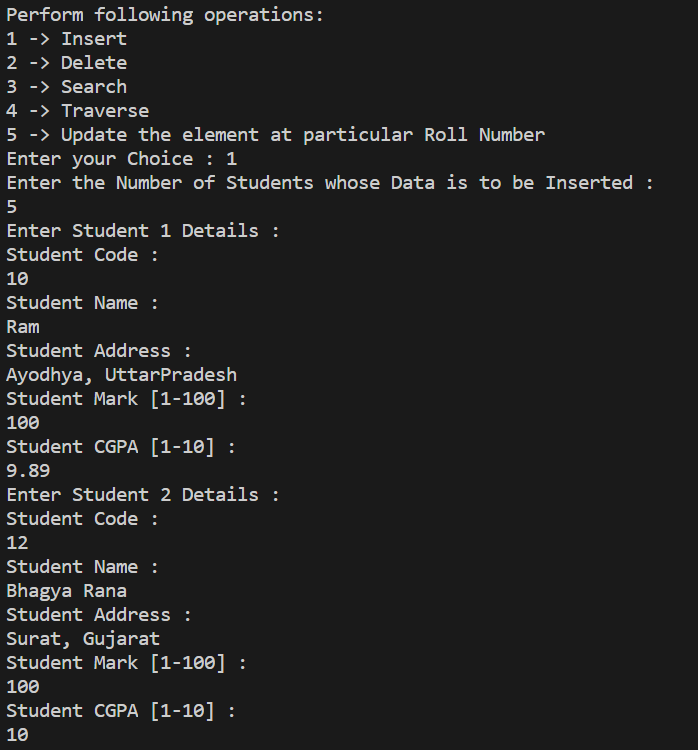
*return* 0;

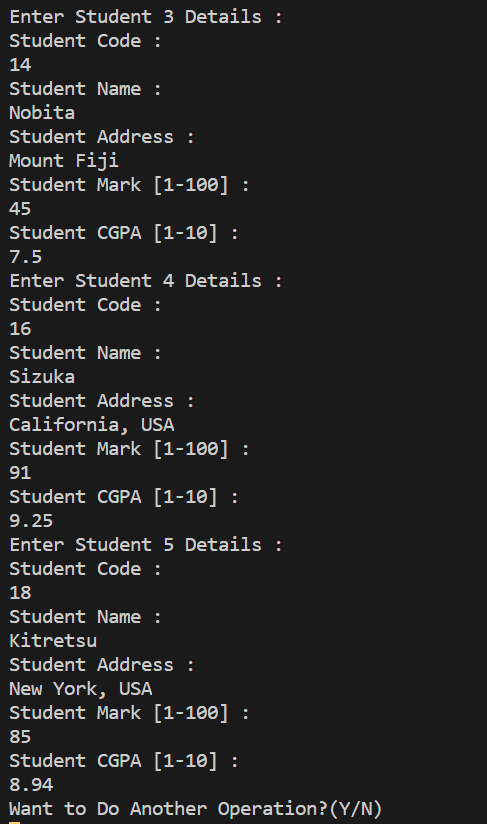
}

Test Case:

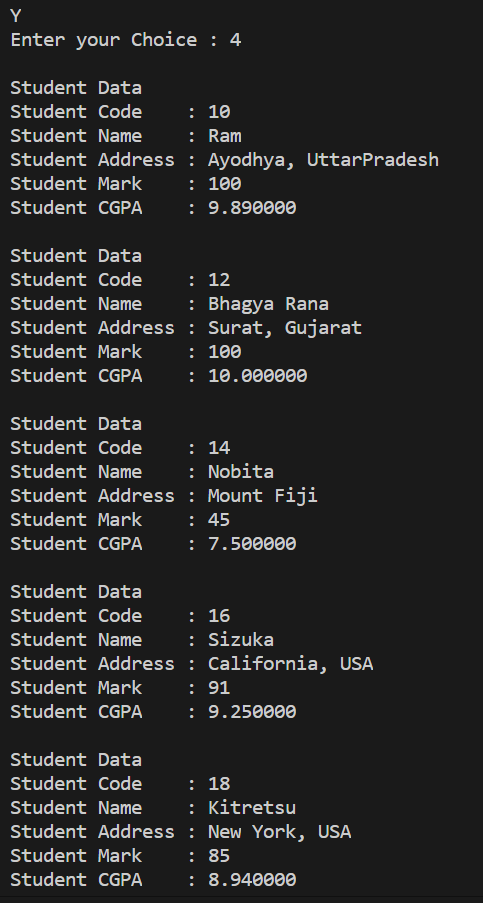
1.) *Inserting* Data of 5 Students

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Sr No | Code | Name | Address | Mark | CGPA |
| 1 | 10 | Ram | Ayodhya, UttarPradesh | 100 | 9.89 |
| 2 | 12 | Bhagya Rana | Surat, Gujarat | 100 | 10 |
| 3 | 14 | Nobita | Mount Fiji | 45 | 7.5 |
| 4 | 16 | Sizuka | California, USA | 91 | 9.25 |
| 5 | 18 | Kitretsu | New York, USA | 85 | 8.94 |



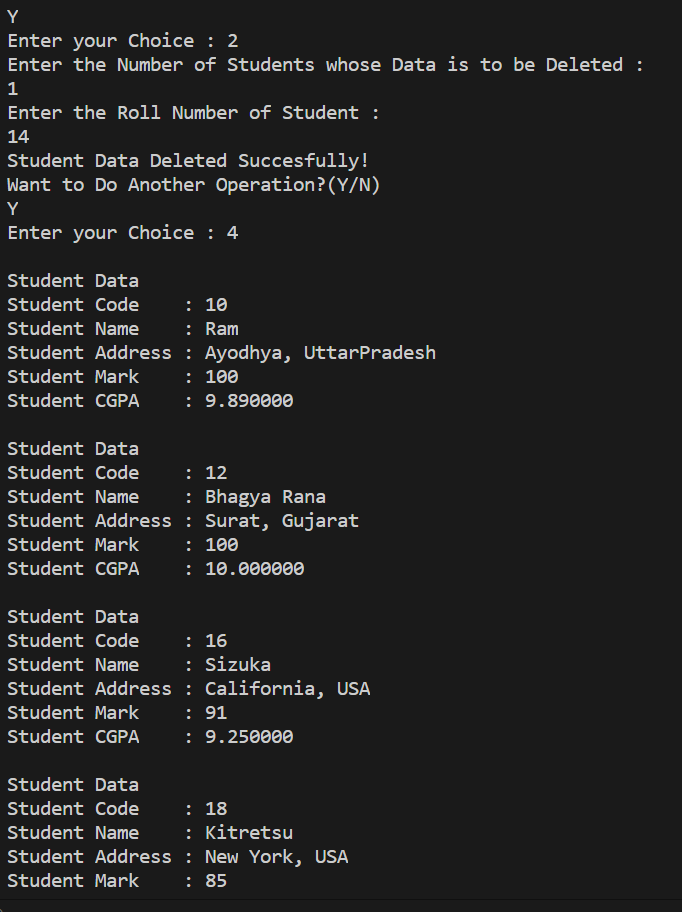


2.) *Traversal* of all Students Data: [To Confirm Data Has been Entered Correctly!]

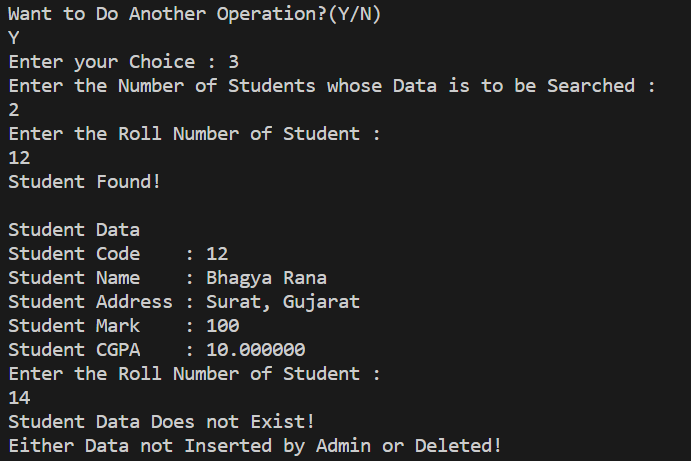


3.) *Deleting Nobita* [Roll Number: 14] from Student Data.

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Sr No | Code | Name | Address | Mark | CGPA |
| 1 | 10 | Ram | Ayodhya, UttarPradesh | 100 | 9.89 |
| 2 | 12 | Bhagya Rana | Surat, Gujarat | 100 | 10 |
| 3 | 16 | Sizuka | California, USA | 91 | 9.25 |
| 4 | 18 | Kitretsu | New York, USA | 85 | 8.94 |



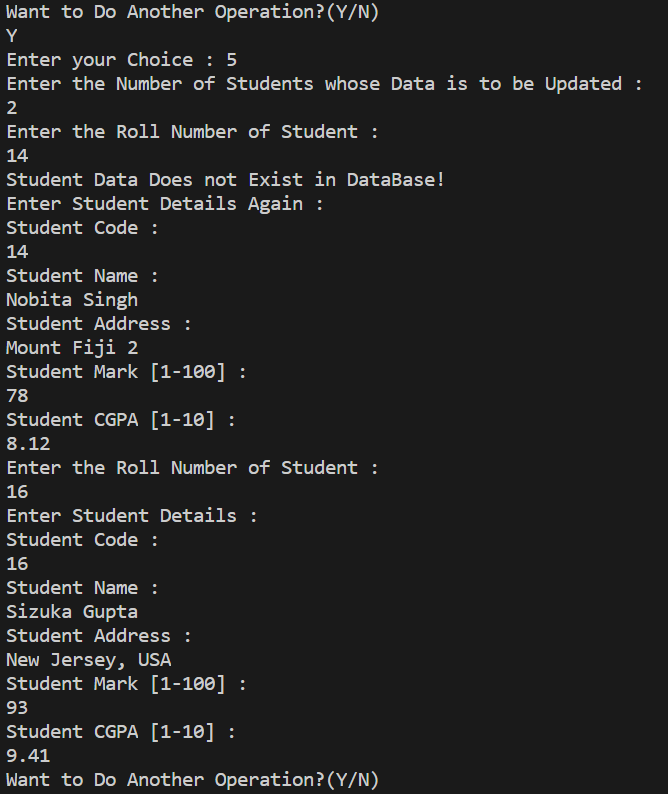
4.) *Search* for Roll Number 12 [Me!] & Roll Number 14 [Nobita] {We deleted Him Recently}

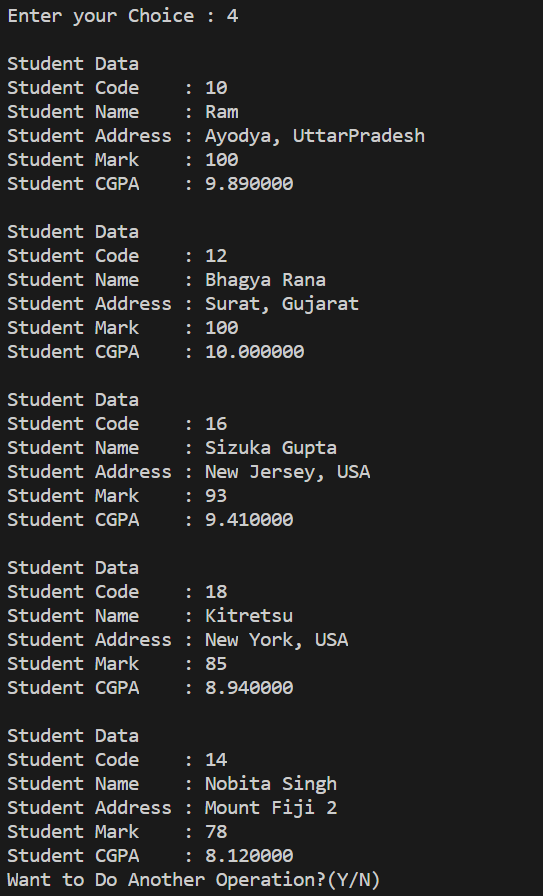


5.) Updating of Table Data

Updated Shizuka Data and Inserted Nobita Data [In Case Data Not Found, Then New Data Inserted!]

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Sr No | Code | Name | Address | Mark | CGPA |
| 1 | 10 | Ram | Ayodhya, UttarPradesh | 100 | 9.89 |
| 2 | 12 | Bhagya Rana | Surat, Gujarat | 100 | 10 |
| 3 | 16 | Sizuka Gupta | New Jersey, USA | 93 | 9.41 |
| 4 | 18 | Kitretsu | New York, USA | 85 | 8.94 |
| 5 | 14 | Nobita Singh | Mount Fiji 2 | 78 | 8.12 |

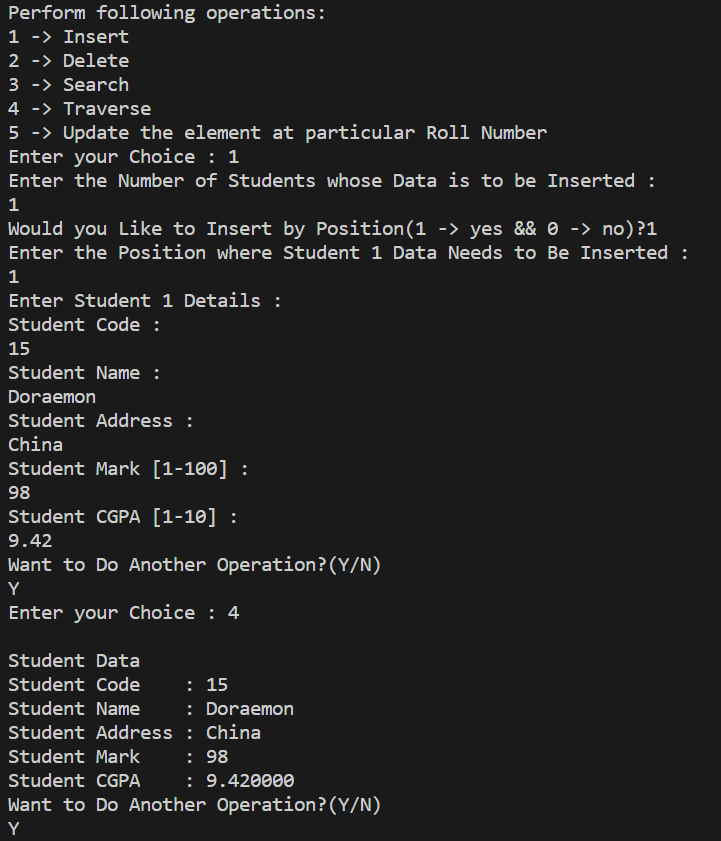




UPDATE: INSERTION AND DELETION LOGIC ADDED

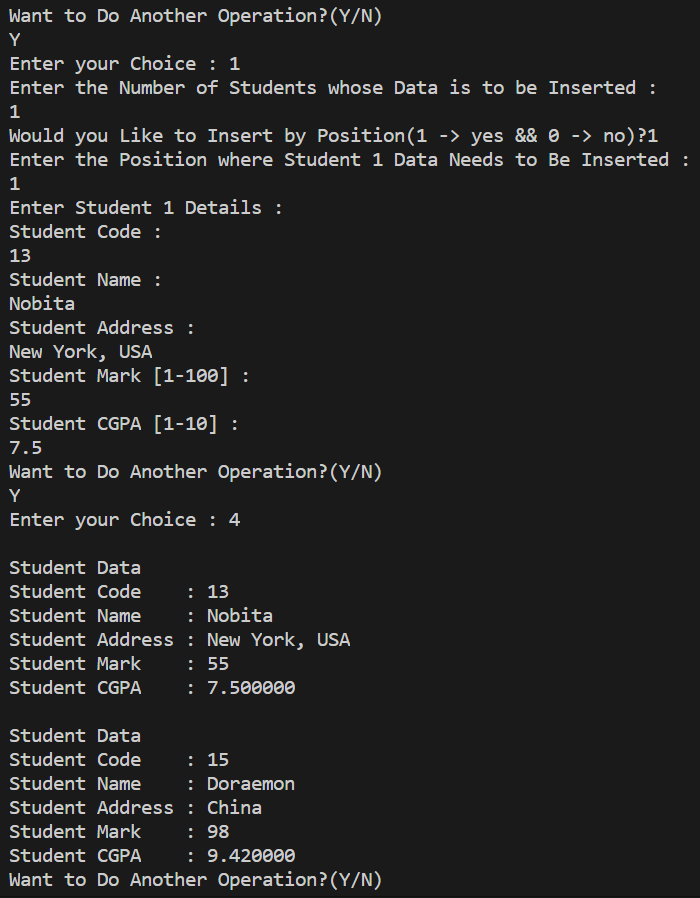
1.) Inserted One Student Data at Position 1

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Position | Code | Name | Address | Mark | CGPA |
| 1 | 15 | Doraemon | China | 98 | 9.42 |



2.) Now, Will Insert Another Student Data at Position, Shifting Old Position One Data.

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Position | Code | Name | Address | Mark | CGPA |
| 1 | 13 | Nobita | New York, USA | 55 | 7.5 |
| 2 | 15 | Doraemon | China | 98 | 9.42 |



Submitted By:

Roll Number: **U19CS012** (*D-12*)

Name: *Bhagya Rana*