

**Learn IN Depth**

Be Professional in Embedded System



# Student Management System

---

- **Name:** Mohamed Mohsen
- **Unit:** 5 – Project 2

## Problem Statement

A simple software used for managing students' information by performing the following operations:

- 1) Reading students' information either manually or from a file.
- 2) Display information of a student using roll number.
- 3) Display information of students using first name.
- 4) Display information of students using course ID.
- 5) Display the total number of students.
- 6) Update student information using roll number.
- 7) Delete student information using roll number.
- 8) Display all students' information.

## Approach

- Each requirement is implemented using a function that performs the functionality required supported by guiding messages to inform the user what to do.
  - 1) Read student data manually.
  - 2) Read students' data from a file.
  - 3) Print information of student using Roll Number.
  - 4) Print information of students using First Name
  - 5) Print information of students using course ID.
  - 6) Display total number of students.
  - 7) Update student information using roll number.
  - 8) Delete student information using roll number.
  - 9) Display all students' information.
- Student data are stored in a structure data type with five attributes.

- 1) Roll Number : integer type
- 2) First Name : character array type
- 3) Last Name : character array type
- 4) GPA : float type
- 5) Courses' ID : integer array type

## Files

This project consists of 7 files:

- main.c
- fifo.c
- fifo.h
- data.txt
- types.h
- student\_ms.c
- student\_ms.h

## main.c

The students' data are stored in a queue data structure where each element is of type structure student.

In this file user choices are taken as input, and according to the user choice the switch statement will call the suitable function to perform the task.

- Initialization of student data base using FIFO with 50 elements.

```
FIFO_init(&S_student_FIFO, S_students_buff, 50);
int choice = 0 ;
int num ;
char f_name[50];
char file_path[255] = "data.txt";
```

- - Declaration of other variables required in the process.
- Declaration and definition of the function get\_choice() which prompts the user with the right choices and returns their choice

```
int get_user_choice(void){
    int return_value = 0 ;
    printf("=====\n");
    printf("Choose The Task you want to perform\n");
    printf("1) Add Student Data Manually\n");
    printf("2) Add Student Data From a Text File\n");
    printf("3) Get Student Details by Roll Number\n");
    printf("4) Get Student Details by First Name\n");
    printf("5) Get Students' Details by Course Number\n");
    printf("6) Get Total Number of Students\n");
    printf("7) Delete The Student Details by Roll Number\n");
    printf("8) Update The Student Details by Roll Number\n");
    printf("9) Show all information\n");
    printf("10) To Exit\n");
    printf("Enter Your Choice To Perform The Task: ");
    fflush(stdin); fflush(stdout);
    scanf("%d", &return_value);

    return return_value ;
}
```

- - Using a do while loop which will continue to iterate until the user's choice is equal to 10.

- Inside this loop we perform a condition on the user's choice using switch statement to prompt them the required output and call the function which will perform the task.

```
printf("Welcome To The Student Management System\n");
do{
    choice = get_user_choice() ;
    switch(choice){
    case 1:{
        /* Add Student Data Manually */
        FIFO_DATA_TYPE item ;
        printf("-----\n");
        printf("Add the student details\n");
        printf("-----\n");
        printf("Enter The Roll Number: ");
        fflush(stdin); fflush(stdout);
        scanf("%d", &(item.roll));

        printf("Enter The First Name: ");
        fflush(stdin); fflush(stdout);
        gets(item.first_name);

        printf("Enter The Last Name: ");
        fflush(stdin); fflush(stdout);
        gets(item.last_name);

        printf("Enter The GPA: ");
        fflush(stdin); fflush(stdout);
        scanf("%f", &(item.gpa));

        printf("Enter The Course ID of each Course: \n");
        for(int i=1; i<=5; i++){
            printf("Course %d ID: ", i);
            fflush(stdin); fflush(stdout);
            scanf("%d", &(item.course_id[i-1]));
        }

        SM_add_student_manually(&S_student_FIFO, item);

    }break;
    case 2:{
        /* Add Student Data from a file */
        printf("Enter File Path: ");
        fflush(stdin); fflush(stdout);
        gets(file_path);
        SM_add_student_file(&S_student_FIFO, file_path);
    }break;
```

○

```

case 3:{
    /* Get Student Details by Roll Number */
    printf("-----\n");
    printf("Enter Roll Number: ");
    fflush(stdin); fflush(stdout);
    scanf("%d", &num);
    SM_get_detail_roll(&S_student_FIFO, num);
}break;
case 4:{
    /* Get Student Details by First Name */
    printf("-----\n");
    printf("Enter First Name: ");
    fflush(stdin); fflush(stdout);
    gets(f_name);
    SM_get_detail_f_name(&S_student_FIFO, f_name);
}break;
case 5:{
    /* Get Students' Details by course Number */
    printf("-----\n");
    printf("Enter Course Number: ");
    fflush(stdin); fflush(stdout);
    scanf("%d", &num);
    SM_get_getail_course_id(&S_student_FIFO, num);
}break;
case 6:{
    /* Get Total Number of Students */
    printf("-----\n");
    printf("Total Number of Students is: %ld\n", SM_get_student_count(&S_student_FIFO));
}break;
case 7:{
    /* Delete Student Details Using Roll Number*/
    printf("-----\n");
    printf("Enter Roll Number: ");
    fflush(stdin); fflush(stdout);
    scanf("%d", &num);
    SM_delete_roll(&S_student_FIFO, num);
}break;
}

```

○

```

case 8:{
    /* Update Student Details Using Roll Number*/
    FIFO_DATA_TYPE item ;
    printf("Enter Roll Number:");
    fflush(stdin); fflush(stdout);
    scanf("%d", &num);

    printf("-----\n");
    printf("Add the student details\n");
    printf("-----\n");
    printf("Enter The Roll Number: ");
    fflush(stdin); fflush(stdout);
    scanf("%d", &(item.roll));

    printf("Enter The First Name: ");
    fflush(stdin); fflush(stdout);
    gets(item.first_name);

    printf("Enter The Last Name: ");
    fflush(stdin); fflush(stdout);
    gets(item.last_name);

    printf("Enter The GPA: ");
    fflush(stdin); fflush(stdout);
    scanf("%f", &(item.gpa));

    printf("Enter The Course ID of each Course: \n");
    for(int i=1; i<=5; i++){
        printf("Course %d ID: ", i);
        fflush(stdin); fflush(stdout);
        scanf("%d", &(item.course_id[i-1]));
    }
    SM_update_roll(&S_student_FIFO, num, item);

}break;
case 9:{
    /* Show all information */
    SM_show(&S_student_FIFO);

}break;
}
}while(choice != 10);

```

○

## student\_ms.h

In this file:

- Global variables used in the process are defined:
  - S\_student\_buff which is an array of 50 elements of type struct S\_Student
  - S\_student\_FIFO which is of type FIFO\_buff\_t, and is to keep track of information about the FIFO like length, count, head, tail, and base.
- Function declaration of functions used to perform the tasks required by the user.

```
#ifndef STUDENT_MS_H
#define STUDENT_MS_H

#include "fifo.h"
#include "types.h"
#include <stdio.h>
#include <string.h>
#include <stdlib.h>

#define LINE_SIZE 255

/* Students */
FIFO_buff_t S_student_FIFO ;
S_student S_students_buff[50];

/* API */
int SM_roll_exists(int r);
void SM_add_student_manually(FIFO_buff_t* fifo, FIFO_DATA_TYPE item);
void SM_add_student_file(FIFO_buff_t* fifo, char file_path[]);
void SM_get_detail_roll(FIFO_buff_t* fifo, int roll_n);
void SM_get_detail_f_name(FIFO_buff_t* fifo, char f_name[]);
void SM_get_getail_course_id(FIFO_buff_t* fifo, int course_n);
unsigned Long SM_get_student_count(FIFO_buff_t* fifo);
void SM_delete_roll(FIFO_buff_t* fifo, int roll_n);
void SM_update_roll(FIFO_buff_t* fifo, int roll_n, FIFO_DATA_TYPE item);
void SM_show(FIFO_buff_t* fifo);

#endif /* STUDENT_MS_H */
```



# student\_ms.c

This file contains function definitions

## 1) SM\_roll\_exists

- This function is used to check whether a roll number exists or not.
- Inputs:
  - The Roll Number : <r> integer
- Outputs:
  - <return\_value> integer
  - Returns 1 if the roll number exists, 0 otherwise.

```
int SM_roll_exists(int r){
    S_student* temp = S_student_FIFO.head ;
    int return_value = 0 ;
    for(int i=0; i<S_student_FIFO.length; i++){
        if(temp == (S_student_FIFO.base + S_student_FIFO.length - 1)){
            temp = S_student_FIFO.base ;
        }else{
            temp ++ ;
        }
        if(temp->roll == r){
            return_value = 1 ;
        }
    }

    return return_value ;
}
```

•

## 2) SM\_add\_student\_manually

- This function is used to add student data manually.
  - Adds a student if the roll number is new.
  - Rejects the operation if the roll number exists.
- Inputs:
  - A pointer to the FIFO
    - <fifo> FIFO\_buff\_t\* type
  - Student data to be added
    - <item> FIFO\_DATA\_TYPE
- Outputs: No output

```
void SM_add_student_manually(FIFO_buff_t* fifo, FIFO_DATA_TYPE item){
    /* Check if the roll number is duplicated ***** */
    if(SM_roll_exists(item.roll)){
        printf("[ERROR] Repeated Roll Number\n");
    }else{
        FIFO_enqueue(fifo, item);
        printf("[INFO] Student Details are added successfully\n");
        printf("-----\n");
        printf("[INFO] Total Number of Students: %ld\n", fifo->count);
        printf("[INFO] You can add up to %ld Students\n", fifo->length);
        printf("[INFO] You can add %ld more Students\n", fifo->length - fifo->count);
        printf("-----\n");
    }
}
```

•

### 3) SM\_add\_student\_file

- This function is used to add students' data from a file.
- Inputs:
  - A pointer to the FIFO : <fifo> FIFO\_buff\_t\* type
  - The file path : <file\_path> pointer to character type
- Outputs: No outputs

```
1 Mohamed Mohsen 3.9 1 2 3 4 5
2 Ali Hossam 4.0 7 8 9 10 11
3 Sayed Hamed 3.8 11 7 5 13 3
10 Mustafa Mandour 3.7 15 1 4 8 2
```

```
void SM_add_student_file(FIFO_buff_t* fifo, char file_path[]){
    FIFO_DATA_TYPE item ;
    FILE * fpointer = NULL ;
    char line[LINE_SIZE] ;
    char* piece = NULL ;
    int i = 0 ;

    fpointer = fopen(file_path, "r");
    while(!feof(fpointer)){
        fgets(line, LINE_SIZE, fpointer);
        /* Split line by space */
        piece = strtok(line, " ");
        i = 0 ;
        while(piece != NULL){
            switch(i){
                case 0:{
                    printf("Roll Number: %s\n", piece);
                    item.roll = atoi(piece);
                }break;
                case 1:{
                    printf("First Name: %s\n", piece);
                    strcpy(item.first_name, piece);
                }break;
                case 2:{
                    printf("Last Name: %s\n", piece);
                    strcpy(item.last_name, piece);
                }break;
                case 3:{
                    printf("GPA: %s\n", piece);
                    item.gpa = atof(piece);
                }break;
                default:{
                    if(i<9){
                        printf("Course %d ID: %s\n", i-3, piece);
                        item.course_id[i-4] = atoi(piece);
                    }
                }break;
            }
            i ++ ;
            piece = strtok(NULL, " ");
        }
        SM_add_student_manually(fifo, item);
    }
    fclose(fpointer);
}
```

#### 4) SM\_get\_detail\_roll

- This function is used to display information about student using roll number
- Inputs:
  - A pointer to FIFO : <fifo> FIFO\_buff\_t\* type
  - Roll Number : <roll\_n> integer type
- Outputs: No Outputs

```
void SM_get_detail_roll(FIFO_buff_t* fifo, int roll_n){
    FIFO_DATA_TYPE* temp = fifo->tail ;
    int j = -1 ;
    printf("==== Roll Number = %d =====\n", roll_n);
    for(int i=0; i<fifo->count; i++){
        if(temp->roll == roll_n){
            printf("=====\n");
            printf("Roll Number: %d\n", temp->roll);
            printf("First Name: %s\n", temp->first_name);
            printf("Last Name: %s\n", temp->last_name);
            printf("GPA: %f\n", temp->gpa);
            for(j=1; j<=5; j++){
                printf("Course %d ID:%d \n", j, temp->course_id[j-1]);
            }
        }
        if(temp == (fifo->base + fifo->length - 1)){
            temp = fifo->base ;
        }else{
            temp ++ ;
        }
    }
    if(j == -1){
        printf("[ERROR] Roll Number Not Found!\n");
    }
}
```

•

## 5) SM\_get\_detail\_f\_name

- This function is used to display students' information using first name.
- Inputs:
  - A pointer to FIFO :<fifo> FIFO\_buff\_t\* type
  - First name: <f\_name> pointer to character type
- Outputs: No outputs

```
void SM_get_detail_f_name(FIFO_buff_t* fifo, char f_name[]){
    FIFO_DATA_TYPE* temp = fifo->tail ;
    int j = -1 ;
    printf("==== First Name = %s =====\n", f_name);
    for(int i=0; i<fifo->count; i++){
        if(strcmp(temp->first_name, f_name) == 0){
            printf("=====\n");
            printf("Roll Number: %d\n", temp->roll);
            printf("First Name: %s\n", temp->first_name);
            printf("Last Name: %s\n", temp->last_name);
            printf("GPA: %f\n", temp->gpa);
            for(j=1; j<=5; j++){
                printf("Course %d ID:%d \n", j, temp->course_id[j-1]);
            }
            if(temp == (fifo->base + fifo->length - 1)){
                temp = fifo->base ;
            }else{
                temp ++ ;
            }
        }
    }
    if(j == -1){
        printf("[ERROR] First Name Not Found!\n");
    }
}
```

•

## 6) SM\_get\_detail\_course\_id

- This function is used to display students' information using course ID.
- Inputs:
  - A pointer to FIFO : <fifo> FIFO\_buff\_t\* type
  - Course Number : <course\_n> integer type
- Outputs: No outputs

```
void SM_get_detail_course_id(FIFO_buff_t* fifo, int course_n){
    FIFO_DATA_TYPE* temp = fifo->tail ;
    int j = -1 ;
    printf("==== Course ID = %d =====\n", course_n);
    for(int i=0; i<fifo->count; i++){
        for(int k=0; k<5; k++){
            if(temp->course_id[k] == course_n){
                printf("=====\n");
                printf("Roll Number: %d\n", temp->roll);
                printf("First Name: %s\n", temp->first_name);
                printf("Last Name: %s\n", temp->last_name);
                printf("GPA: %f\n", temp->gpa);
                for(j=1; j<=5; j++){
                    printf("Course %d ID:%d \n", j, temp->course_id[j-1]);
                }
            }
        }
        if(temp == (fifo->base + fifo->length - 1)){
            temp = fifo->base ;
        }else{
            temp ++ ;
        }
    }
    if(j == -1){
        printf("[ERROR] Course ID Not Found!\n");
    }
}
```

•

## 7) SM\_get\_student\_count

- This function is used to return the number of students in the data base
- Inputs:
  - A pointer to FIFO:<fifo> FIFO\_buff\_t\* type
- Outputs:
  - The number of students in the data base

```
unsigned long SM_get_student_count(FIFO_buff_t* fifo){  
    return fifo->count ;  
}
```

-

## 8) SM\_delete\_roll

- This function is used to delete a student from the data base using their roll number
- Inputs :
  - A pointer to FIFO : <fifo> FIFO\_buff\_t\* type
  - Roll number : <roll\_n> integer type
- Outputs: No outputs

```
void SM_delete_roll(FIFO_buff_t* fifo, int roll_n){
    FIFO_DATA_TYPE* temp = fifo->tail ;
    FIFO_DATA_TYPE* roll_ptr = NULL /* ((void*)0)*/ ;
    for(int i=1; i<=fifo->count; i++){
        if(temp->roll == roll_n){
            roll_ptr = temp ;
        }
        if(temp == (fifo->base + fifo->length - 1)){
            temp = fifo->base ;
        }else{
            temp ++ ;
        }
    }

    if(roll_ptr == NULL){
        printf("[ERROR] Roll Number Not Found!\n");
    }else{
        temp = roll_ptr + 1 ;
        if(temp > (fifo->base + fifo->length - 1)){
            temp = fifo->base ;
        }
        while(temp != fifo->head){
            *((FIFO_DATA_TYPE*)(roll_ptr)) = *((FIFO_DATA_TYPE*)(temp)) ;

            if(temp == (fifo->base + fifo->length - 1)){
                temp = fifo->base ;
            }else{
                temp ++ ;
            }
            if(roll_ptr == (fifo->base + fifo->length - 1)){
                roll_ptr = fifo->base ;
            }else{
                roll_ptr ++ ;
            }
        }
        *((FIFO_DATA_TYPE*)(roll_ptr)) = *((FIFO_DATA_TYPE*)(temp)) ;
        fifo->head = roll_ptr;
        fifo->count -- ;
    }
}
```



## 9) SM\_update\_roll

- This function is used to update a student's information using roll number.
- Inputs:
  - A pointer to FIFO : <fifo> FIFO\_buff\_t\* type
  - Roll Number : <roll\_n> integer type
  - Student data to be added : <item> FIFO\_DATA\_TYPE type
- Outputs: No outputs

```
void SM_update_roll(FIFO_buff_t* fifo, int roll_n, FIFO_DATA_TYPE item){
    FIFO_DATA_TYPE* temp = fifo->tail ;
    FIFO_DATA_TYPE* roll_ptr = NULL /* ((void*)0)*/ ;
    for(int i=1; i<fifo->count; i++){
        if(temp->roll == roll_n){
            roll_ptr = temp ;
        }
        if(temp == (fifo->base + fifo->length - 1)){
            temp = fifo->base ;
        }else{
            temp ++ ;
        }
    }

    if(roll_ptr == NULL){
        printf("[ERROR] Roll Number Not Found\n");
    }else if(SM_roll_exists(item.roll)){
        printf("[ERROR] Repeated Roll Number\n");
    }else{
        *((FIFO_DATA_TYPE*)(roll_ptr)) = item ;
    }
}
```

•

## 10) SM\_show

- This function is used to show all students' information.
- Inputs: A pointer to FIFO <fifo> FIFO\_buff\_t\* type
- Outputs: No outputs

```
void SM_show(FIFO_buff_t* fifo){
    FIFO_DATA_TYPE* temp = fifo->tail ;
    for(int i=0; i<fifo->count; i++){
        printf("=====\n");
        printf("Roll Number: %d\n", temp->roll);
        printf("First Name: %s\n", temp->first_name);
        printf("Last Name: %s\n", temp->last_name);
        printf("GPA: %f\n", temp->gpa);
        for(int j=1; j<=5; j++){
            printf("Course %d ID:%d \n", j, temp->course_id[j-1]);
        }
        if( temp == (fifo->base + fifo->length - 1)){
            temp = fifo->base ;
        }else{
            temp ++ ;
        }
    }
}
```

•

# Program Execution

```
Welcome To The Student Management System
=====
Choose The Task you want to perform
1) Add Student Data Manually
2) Add Student Data From a Text File
3) Get Student Details by Roll Number
4) Get Student Details by First Name
5) Get Students' Details by Course Number
6) Get Total Number of Students
7) Delete The Student Details by Roll Number
8) Update The Student Details by Roll Number
9) Show all information
10) To Exit
Enter Your Choice To Perform The Task: 2
Enter File Path: data.txt
Roll Number: 1
First Name: Mohamed
Last Name: Mohsen
GPA: 3.9
Course 1 ID: 1
Course 2 ID: 2
Course 3 ID: 3
Course 4 ID: 4
Course 5 ID: 5

[INFO] Student Details are added successfully
-----
[INFO] Total Number of Students: 1
[INFO] You can add up to 50 Students
[INFO] You can add 49 more Students
-----
Roll Number: 2
First Name: Ali
Last Name: Hossam
GPA: 4.0
Course 1 ID: 7
Course 2 ID: 8
Course 3 ID: 9
Course 4 ID: 10
Course 5 ID: 11

[INFO] Student Details are added successfully
-----
[INFO] Total Number of Students: 2
[INFO] You can add up to 50 Students
[INFO] You can add 48 more Students
```

```

-----
Roll Number: 3
First Name: Sayed
Last Name: Hamed
GPA: 3.8
Course 1 ID: 11
Course 2 ID: 7
Course 3 ID: 5
Course 4 ID: 13
Course 5 ID: 3

[INFO] Student Details are added successfully
-----
[INFO] Total Number of Students: 3
[INFO] You can add up to 50 Students
[INFO] You can add 47 more Students
-----
Roll Number: 10
First Name: Mustafa
Last Name: Mandour
GPA: 3.7
Course 1 ID: 15
Course 2 ID: 1
Course 3 ID: 4
Course 4 ID: 8
Course 5 ID: 2
[INFO] Student Details are added successfully
-----
[INFO] Total Number of Students: 4
[INFO] You can add up to 50 Students
[INFO] You can add 46 more Students
-----
=====
Choose The Task you want to perform
1) Add Student Data Manually
2) Add Student Data From a Text File
3) Get Student Details by Roll Number
4) Get Student Details by First Name
5) Get Students' Details by Course Number
6) Get Total Number of Students
7) Delete The Student Details by Roll Number
8) Update The Student Details by Roll Number
9) Show all information
10) To Exit
Enter Your Choice To Perform The Task: 1
-----
Add the student details
-----

```

```
Enter The Roll Number: 4
Enter The First Name: Baseem
Enter The Last Name: Saleh
Enter The GPA: 3.2
Enter The Course ID of each Course:
Course 1 ID: 5
Course 2 ID: 4
Course 3 ID: 17
Course 4 ID: 2
Course 5 ID: 6
[INFO] Student Details are added successfully
-----
[INFO] Total Number of Students: 5
[INFO] You can add up to 50 Students
[INFO] You can add 45 more Students
-----
=====
Choose The Task you want to perform
1) Add Student Data Manually
2) Add Student Data From a Text File
3) Get Student Details by Roll Number
4) Get Student Details by First Name
5) Get Students' Details by Course Number
6) Get Total Number of Students
7) Delete The Student Details by Roll Number
8) Update The Student Details by Roll Number
9) Show all information
10) To Exit
Enter Your Choice To Perform The Task: 1
-----
Add the student details
-----
Enter The Roll Number: 10
Enter The First Name: Ahmad
Enter The Last Name: Mohamed
Enter The GPA: 3.5
Enter The Course ID of each Course:
Course 1 ID: 1
Course 2 ID: 2
Course 3 ID: 3
Course 4 ID: 4
Course 5 ID: 5
[ERROR] Repeated Roll Number
```

```

=====
Choose The Task you want to perform
1) Add Student Data Manually
2) Add Student Data From a Text File
3) Get Student Details by Roll Number
4) Get Student Details by First Name
5) Get Students' Details by Course Number
6) Get Total Number of Students
7) Delete The Student Details by Roll Number
8) Update The Student Details by Roll Number
9) Show all information
10) To Exit
Enter Your Choice To Perform The Task: 3
-----
Enter Roll Number: 10
===== Roll Number = 10 =====
=====
Roll Number: 10
First Name: Mustafa
Last Name: Mandour
GPA: 3.700000
Course 1 ID:15
Course 2 ID:1
Course 3 ID:4
Course 4 ID:8
Course 5 ID:2
=====

```

```

=====
Choose The Task you want to perform
1) Add Student Data Manually
2) Add Student Data From a Text File
3) Get Student Details by Roll Number
4) Get Student Details by First Name
5) Get Students' Details by Course Number
6) Get Total Number of Students
7) Delete The Student Details by Roll Number
8) Update The Student Details by Roll Number
9) Show all information
10) To Exit
Enter Your Choice To Perform The Task: 4
-----
Enter First Name: Mohamed
===== First Name = Mohamed =====
=====
Roll Number: 1
First Name: Mohamed
Last Name: Mohsen
GPA: 3.900000
Course 1 ID:1
Course 2 ID:2
Course 3 ID:3
Course 4 ID:4
Course 5 ID:5
=====

```

```

=====
Choose The Task you want to perform
1) Add Student Data Manually
2) Add Student Data From a Text File
3) Get Student Details by Roll Number
4) Get Student Details by First Name
5) Get Students' Details by Course Number
6) Get Total Number of Students
7) Delete The Student Details by Roll Number
8) Update The Student Details by Roll Number
9) Show all information
10) To Exit
Enter Your Choice To Perform The Task: 5
-----
Enter Course Number: 1
===== Course ID = 1 =====
=====
Roll Number: 1
First Name: Mohamed
Last Name: Mohsen
GPA: 3.900000
Course 1 ID:1
Course 2 ID:2
Course 3 ID:3
Course 4 ID:4
Course 5 ID:5
=====
Roll Number: 10
First Name: Mustafa
Last Name: Mandour
GPA: 3.700000
Course 1 ID:15
Course 2 ID:1
Course 3 ID:4
Course 4 ID:8
Course 5 ID:2
=====

```

```

=====
Choose The Task you want to perform
1) Add Student Data Manually
2) Add Student Data From a Text File
3) Get Student Details by Roll Number
4) Get Student Details by First Name
5) Get Students' Details by Course Number
6) Get Total Number of Students
7) Delete The Student Details by Roll Number
8) Update The Student Details by Roll Number
9) Show all information
10) To Exit
Enter Your Choice To Perform The Task: 6
-----
Total Number of Students is: 5

```

```
=====
Choose The Task you want to perform
1) Add Student Data Manually
2) Add Student Data From a Text File
3) Get Student Details by Roll Number
4) Get Student Details by First Name
5) Get Students' Details by Course Number
6) Get Total Number of Students
7) Delete The Student Details by Roll Number
8) Update The Student Details by Roll Number
9) Show all information
10) To Exit
Enter Your Choice To Perform The Task: 8
Enter Roll Number:10
-----
Add the student details
-----
Enter The Roll Number: 11
Enter The First Name: Mostafa
Enter The Last Name: Hussein
Enter The GPA: 3.9
Enter The Course ID of each Course:
Course 1 ID: 1
Course 2 ID: 5
Course 3 ID: 8
Course 4 ID: 9
Course 5 ID: 17
```



```
=====
Choose The Task you want to perform
1) Add Student Data Manually
2) Add Student Data From a Text File
3) Get Student Details by Roll Number
4) Get Student Details by First Name
5) Get Students' Details by Course Number
6) Get Total Number of Students
7) Delete The Student Details by Roll Number
8) Update The Student Details by Roll Number
9) Show all information
10) To Exit
Enter Your Choice To Perform The Task: 9
=====
Roll Number: 1
First Name: Mohamed
Last Name: Mohsen
GPA: 3.900000
Course 1 ID:1
Course 2 ID:2
Course 3 ID:3
Course 4 ID:4
Course 5 ID:5
=====
Roll Number: 2
First Name: Ali
Last Name: Hossam
GPA: 4.000000
Course 1 ID:7
Course 2 ID:8
Course 3 ID:9
Course 4 ID:10
Course 5 ID:11
=====
Roll Number: 3
First Name: Sayed
Last Name: Hamed
GPA: 3.800000
Course 1 ID:11
Course 2 ID:7
Course 3 ID:5
Course 4 ID:13
Course 5 ID:3
```

```
=====
Roll Number: 11
First Name: Mostafa
Last Name: Hussein
GPA: 3.900000
Course 1 ID:1
Course 2 ID:5
Course 3 ID:8
Course 4 ID:9
Course 5 ID:17
=====
Roll Number: 4
First Name: Baseem
Last Name: Saleh
GPA: 3.200000
Course 1 ID:5
Course 2 ID:4
Course 3 ID:17
Course 4 ID:2
Course 5 ID:6
=====
```

```
=====
Choose The Task you want to perform
1) Add Student Data Manually
2) Add Student Data From a Text File
3) Get Student Details by Roll Number
4) Get Student Details by First Name
5) Get Students' Details by Course Number
6) Get Total Number of Students
7) Delete The Student Details by Roll Number
8) Update The Student Details by Roll Number
9) Show all information
10) To Exit
Enter Your Choice To Perform The Task: 7
-----
Enter Roll Number: 11
=====
```

```

=====
Choose The Task you want to perform
1) Add Student Data Manually
2) Add Student Data From a Text File
3) Get Student Details by Roll Number
4) Get Student Details by First Name
5) Get Students' Details by Course Number
6) Get Total Number of Students
7) Delete The Student Details by Roll Number
8) Update The Student Details by Roll Number
9) Show all information
10) To Exit
Enter Your Choice To Perform The Task: 9
=====
Roll Number: 1
First Name: Mohamed
Last Name: Mohsen
GPA: 3.900000
Course 1 ID:1
Course 2 ID:2
Course 3 ID:3
Course 4 ID:4
Course 5 ID:5
=====
Roll Number: 2
First Name: Ali
Last Name: Hossam
GPA: 4.000000
Course 1 ID:7
Course 2 ID:8
Course 3 ID:9
Course 4 ID:10
Course 5 ID:11
=====
Roll Number: 3
First Name: Sayed
Last Name: Hamed
GPA: 3.800000
Course 1 ID:11
Course 2 ID:7
Course 3 ID:5
Course 4 ID:13
Course 5 ID:3
=====

```

```

=====
Roll Number: 4
First Name: Baseem
Last Name: Saleh
GPA: 3.200000
Course 1 ID:5
Course 2 ID:4
Course 3 ID:17
Course 4 ID:2
Course 5 ID:6

```

```

=====
Choose The Task you want to perform
1) Add Student Data Manually
2) Add Student Data From a Text File
3) Get Student Details by Roll Number
4) Get Student Details by First Name
5) Get Students' Details by Course Number
6) Get Total Number of Students
7) Delete The Student Details by Roll Number
8) Update The Student Details by Roll Number
9) Show all information
10) To Exit
Enter Your Choice To Perform The Task: 7
-----
Enter Roll Number: 4
=====
Choose The Task you want to perform
1) Add Student Data Manually
2) Add Student Data From a Text File
3) Get Student Details by Roll Number
4) Get Student Details by First Name
5) Get Students' Details by Course Number
6) Get Total Number of Students
7) Delete The Student Details by Roll Number
8) Update The Student Details by Roll Number
9) Show all information
10) To Exit
Enter Your Choice To Perform The Task: 9
=====
Roll Number: 1
First Name: Mohamed
Last Name: Mohsen
GPA: 3.900000
Course 1 ID:1
Course 2 ID:2
Course 3 ID:3
Course 4 ID:4
Course 5 ID:5
=====
Roll Number: 2
First Name: Ali
Last Name: Hossam
GPA: 4.000000
Course 1 ID:7
Course 2 ID:8
Course 3 ID:9
Course 4 ID:10
=====

```

```
=====
Roll Number: 3
First Name: Sayed
Last Name: Hamed
GPA: 3.800000
Course 1 ID:11
Course 2 ID:7
Course 3 ID:5
Course 4 ID:13
Course 5 ID:3
=====
Choose The Task you want to perform
1) Add Student Data Manually
2) Add Student Data From a Text File
3) Get Student Details by Roll Number
4) Get Student Details by First Name
5) Get Students' Details by Course Number
6) Get Total Number of Students
7) Delete The Student Details by Roll Number
8) Update The Student Details by Roll Number
9) Show all information
10) To Exit
Enter Your Choice To Perform The Task: 10
```