

Carmen Municipal College

A.Y. 2024- 2025

## STUDENT MANAGEMENT SYSTEM

CC103- COMPUTER PROGRAMMING 2

### MIDTERM PROJECT

SORENIO, JUSTINE

RUIZ, KATHLEEN

#### I. Introduction

The Student Management System is a C program that helps manage student information. It allows you to add new students (name and age) and view the list of all students.

#### II. Function Description

##### 1. displayMenu()

Purpose: Shows the menu with options to add a student, display all students, or exit.

Inputs: None.

Outputs: Prints the menu to the screen

##### 2. addStudent()

Purpose: Adds a student's name and age to the list.

Inputs: User enters the student's name and age.

Outputs: Stores the details in an array

### 3. displayStudents()

Purpose: Displays all the students in the list.

Inputs: None.

Outputs: Prints the name and age of each student.

### III. Code Implementation

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

```
#include <string.h>
```

```
#define MAX_STUDENTS 100
```

```
struct Student {
```

```
    char name[50];
```

```
    int age;
```

```
};
```

```
// Global variables
```

```
struct Student students[MAX_STUDENTS];
```

```
int studentCount = 0;
```

```
// Function declarations
```

```
void displayMenu();
```

```
void addStudent();
```

```
void displayStudents();
```

```
int main() {  
    int choice;  
  
    while (1) { // Menu loop  
        displayMenu();  
        printf("Enter your choice: ");  
        scanf("%d", &choice);  
  
        switch (choice) {  
            case 1:  
                addStudent();  
                break;  
            case 2:  
                displayStudents();  
                break;  
            case 3:  
                printf("Exiting the program. Goodbye!\n");  
                return 0; // Exit  
            default:  
                printf("Invalid choice. Please try again.\n");  
        }  
    }  
  
    return 0;  
}
```

```
}
```

```
// Function to display the menu
```

```
void displayMenu() {  
    printf("\n=== Student Management System ===\n");  
    printf("1. Add Student\n");  
    printf("2. Display Students\n");  
    printf("3. Exit\n");  
}
```

```
// Function to add a new student
```

```
void addStudent() {  
    if (studentCount < MAX_STUDENTS) {  
        printf("\nEnter student name: ");  
        scanf("%[^\n]", students[studentCount].name); // Read string with spaces  
        printf("Enter student age: ");  
        scanf("%d", &students[studentCount].age);  
        studentCount++;  
        printf("Student added successfully!\n");  
    } else {  
        printf("Student list is full. Cannot add more students.\n");  
    }  
}
```

```
// Function to display all students
```

```

void displayStudents() {
    if (studentCount == 0) {
        printf("\nNo students in the list.\n");
    } else {
        printf("\n=== Student List ===\n");
        for (int i = 0; i < studentCount; i++) {
            printf("Student %d: Name: %s, Age: %d\n", i + 1, students[i].name, students[i].age);
        }
    }
}

```

#### IV. How It Works

##### 1. Menu:

The program starts by showing a menu using the `displayMenu()` function.

The user can select an option to add a student, display the list of students, or exit.

##### 2. Adding a Student:

When the user selects "Add Student," the `addStudent()` function asks for the student's name and age.

The details are stored in an array.

##### 3. Displaying Students:

When the user selects "Display Students," the `displayStudents()` function shows the list of all students added so far.

If no students are added, it informs the user that the list is empty.

##### 4. Repeat or Exit:

After completing an action, the menu is shown again so the user can add more students, view the list, or exit.

## V. Sample Output

### Menu

=== Student Management System ===

1. Add Student
2. Display Students
3. Exit

Enter your choice: 1

### Adding a Student

Enter student name: John Doe

Enter student age: 20

Student added successfully!

### Displaying Students

=== Student List ===

Student 1: Name: John Doe, Age: 20

Student 2: Name: Jane Smith, Age: 22

### Exiting the Program

=== Student Management System ===

1. Add Student
2. Display Students
3. Exit

Enter your choice: 3

Exiting the program.