

Project 2 Learn in Depth

student MAnagement system

Mina Fathy Labib Hakim | Mastering Embedded Systems | 26.08.2024

Table of Contents

[Project Scope / Problem Statement: 3](#_Toc176016099)

[Methodology / Approach: 3](#_Toc176016100)

[Functions Description: 4](#_Toc176016101)

[Software Testing Results: 7](#_Toc176016102)

# Project Scope / Problem Statement:

For this project, a student information management system software has to be implemented using queue method.

The following operations have to be included in the system:

1. Store the first name of the student
2. Store the last name of the student
3. Store the unique roll number of the student
4. Store the GPA of every student
5. Store the courses registered by the student

# Methodology / Approach:

The main idea is to develop different functions for each operation. All functions together will form the entire software. As mentioned earlier, **the mechanism of queue data structure (as a dynamic linked list)** is the underlying methodology behind the entire software.

The entire SW is built up based on the following files:

* Student\_Mangement.c
* Student\_Mangement.h
* main.c
* Platform\_Types.h
* Student\_Mangement.txt

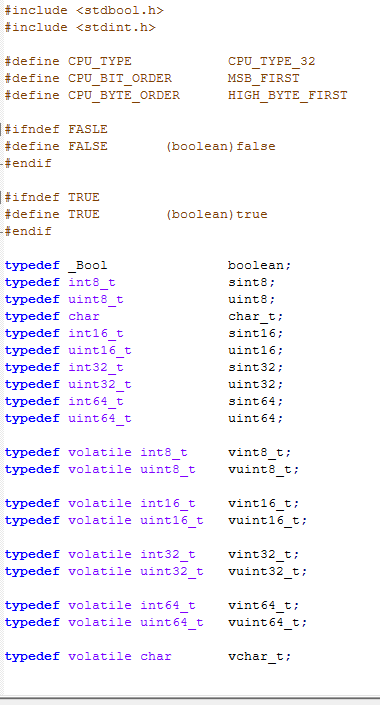
# Functions Description:

**Main.c** is including the main infinite loop with all options to be selected by the user of the software. As well as, Queue\_LinkedList.h is included which provides all functions needed for the software.

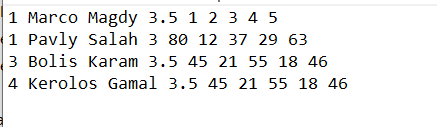
In the **“**Student\_Mangement.h**”** the queue structure is defined along with the declarations of the functions needed to handle the queue mechanism and also the user functions:



In the **“**Platform\_Types.h**”** data types are defined with names that is simplified and indicate the number bits they occupy in the memory.



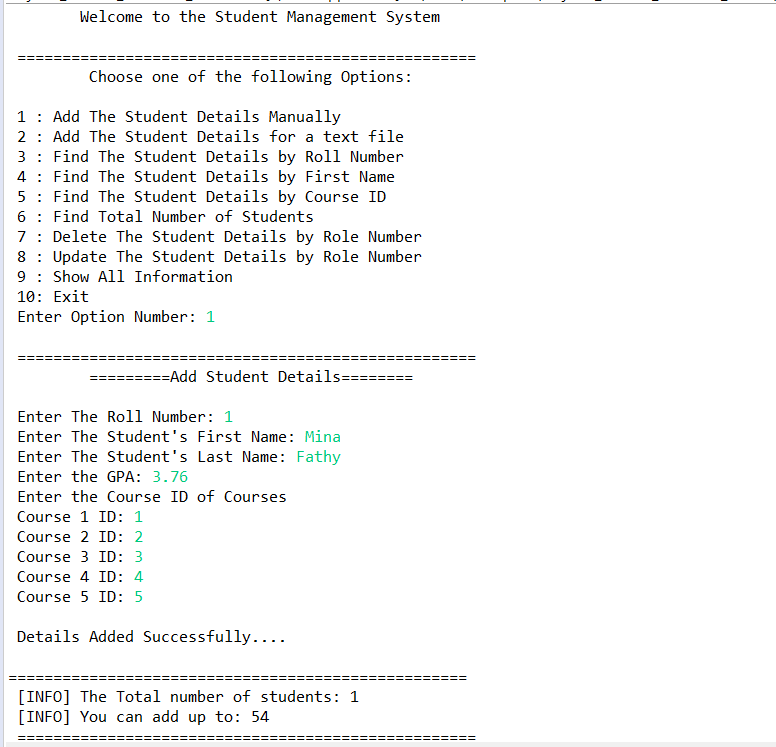
In the **“**Student\_Mangement.txt**”** is previously stored Students data with 2 students having duplicate Roll Numbers on purpose to test system protection against duplicate Roll Numbers.



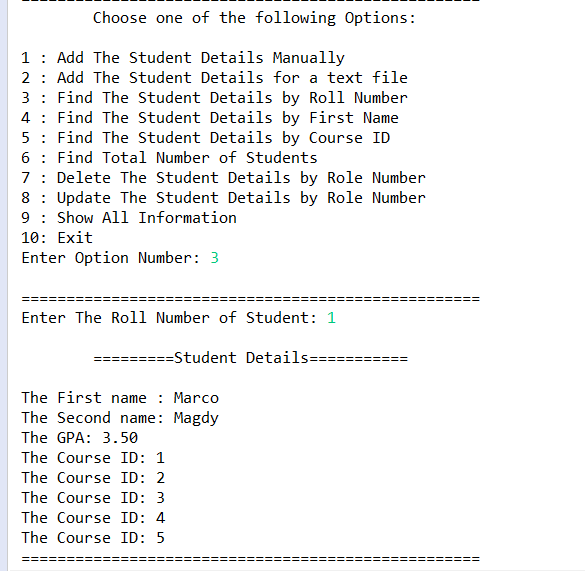
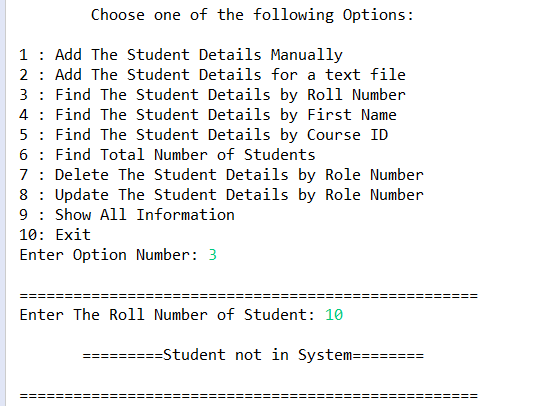
# Software Testing Results:

**Note**: **Students’ Data, used in testing, are from Student\_Mangement.txt, EXCEPT for Add Student Manually test.**

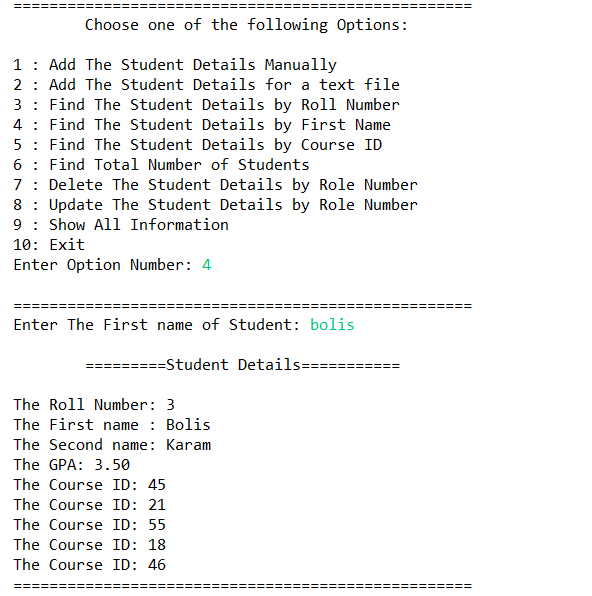
1. Add Students from File:

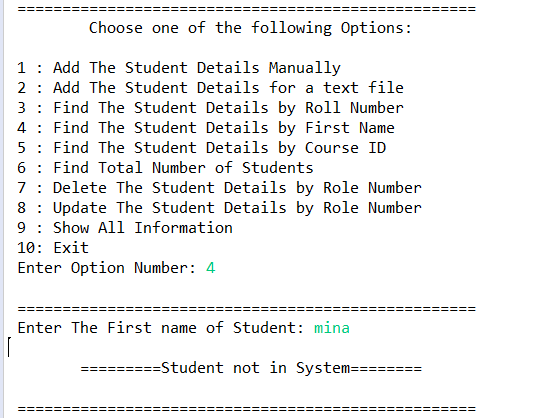


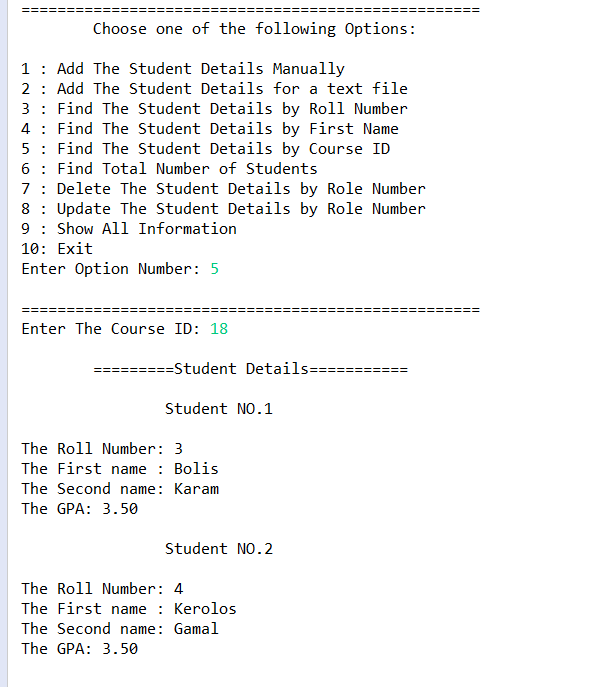
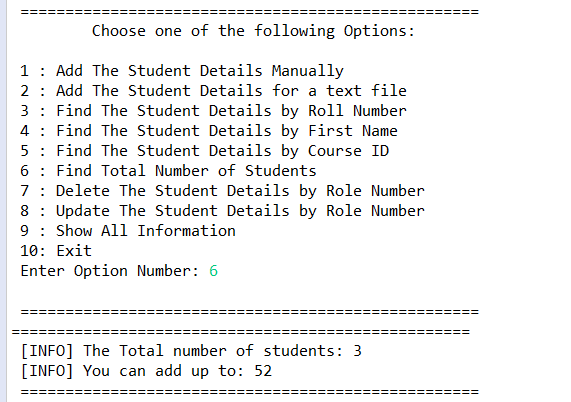
1. Find the student by roll number:

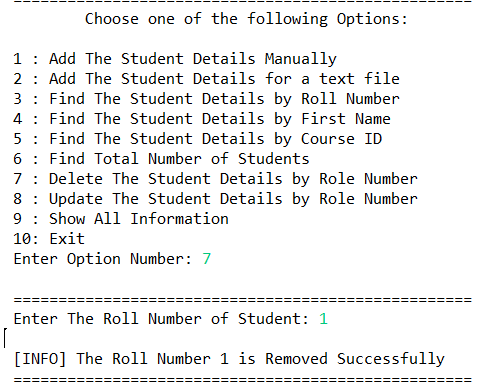
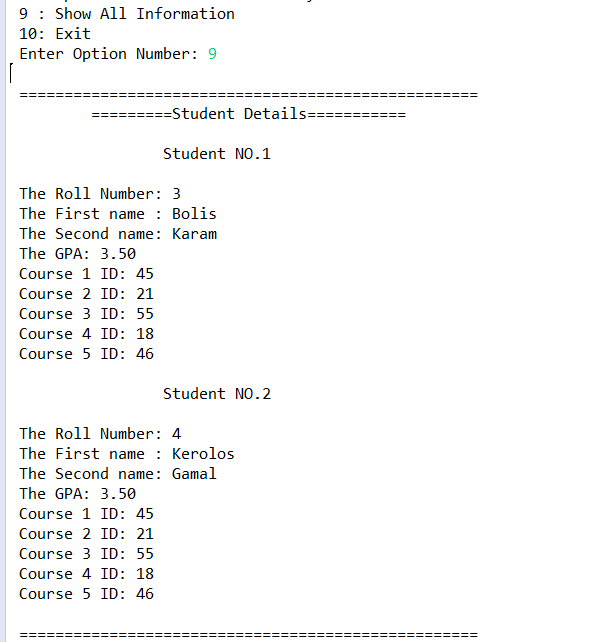
* **Student is Found in System:**
* **Student is not in System:**

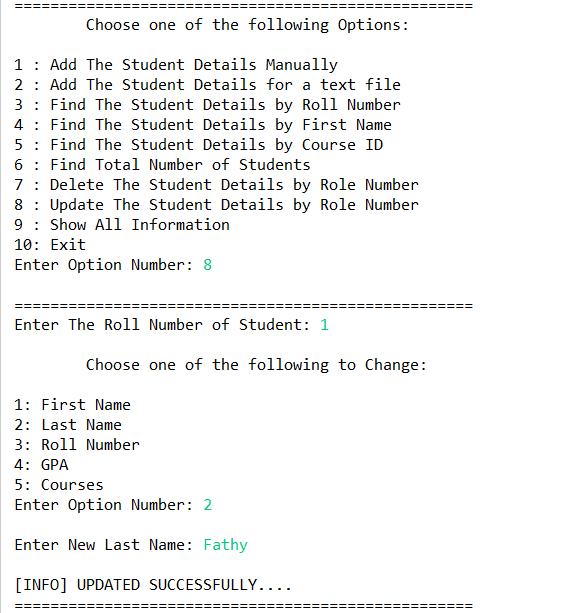
1. Find the student by the first name:

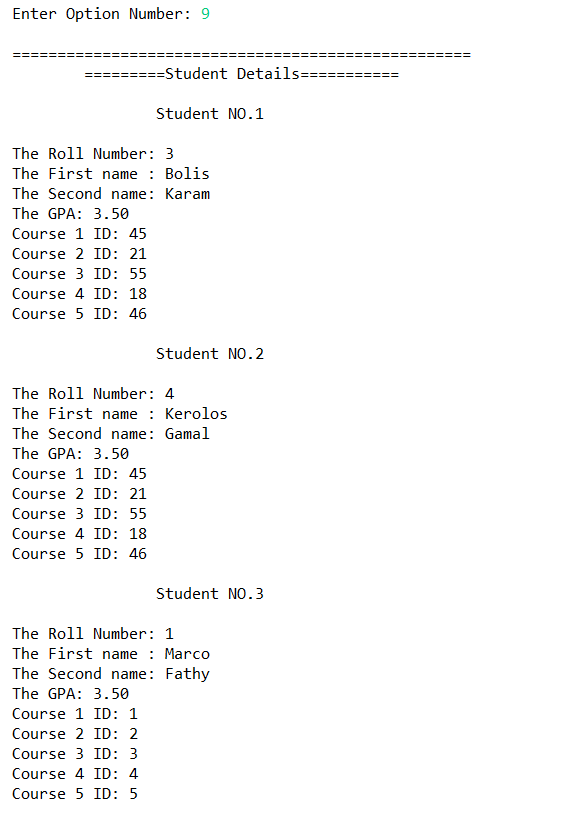
* **Student is Found in System** 
* **Student is not in System**



1. Find all students with the same course ID:
2. Calculate the total number of students:
3. Remove single student by his roll number & display all students:



1. Update single student and display the all students:



Exit the Program:

