**Course: PRF192 - Programming Fundamentals**

**STUDENT MANAGEMENT SYSTEM**

|  |  |
| --- | --- |
| **Group No: 1** | |
| **Group Members** | Dương Thái Bảo - SE190568  Vương Anh Hào - SE183431  Phan Văn Tín - SE190065  Nguyễn Thái Bảo - SE184926 |
| **Lecturer** | Lý Tuấn Anh |
| **Class** | SE2001 |

## Table of Contents

[Table of Contents 2](#_Toc209105045)

[1. Introduction 3](#_Toc209105046)

[1.1. Project Description 3](#_Toc209105047)

[1.2. Project Team 3](#_Toc209105048)

[2. Requirement Analysis & Decomposition 3](#_Toc209105049)

[2.1. Data 3](#_Toc209105050)

[2.2. Functions 3](#_Toc209105051)

[3. Algorithm Design & Flowcharts 4](#_Toc209105052)

[4. Implementation of Basic Functions 5](#_Toc209105053)

[5. Structure Design 6](#_Toc209105054)

[6. File I/O & Testing 7](#_Toc209105055)

[7. Experimental Results 8](#_Toc209105056)

[8. Appendix 14](#_Toc209105057)

## 1. Introduction

### 1.1. Mô tả dự án

### Dự án phát triển Hệ thống Quản lý Sinh viên sử dụng ngôn ngữ lập trình C.

Chương trình cho phép quản lý hồ sơ sinh viên, nhập điểm và tính điểm trung bình (GPA), tìm kiếm, sắp xếp, lưu/đọc dữ liệu từ các tệp, đồng thời áp dụng cấu trúc, mảng cấu trúc.

### 1.2. Project Team

Thông tin nhóm thực hiện dự án

Bảng 1

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **#** | **Student ID** | **Full Name** | **Completion Level** | **Tasks Performed** |
| 1 | SE190568 | Dương Thái Bảo | 90% | Write main code, complete GPA function, testing , report writing |
| 2 | SE183431 | Vương Anh Hào | 90% | Menu, sorting, and search functions |
| 3 | SE190065 | Phan Văn Tín | 90% | Add/Edit/Delete student functions |
| 4 | SE184926 | Nguyễn Thái Bảo | 90% | read/write file and analyze documentation |

## 2. Requirement Analysis & Decomposition

### 2.1. Data

Bảng mô tả các thuộc tính dữ liệu được sử dụng để lưu thông tin sinh viên.

Bảng 2

|  |  |  |
| --- | --- | --- |
| **#** | **Name** | **Description** |
| 1 | Code | Mã sinh viên |
| 2 | Name | Họ và tên sinh viên |
| 3 | BirthYear | Năm sinh |
| 4 | Major | Chuyên ngành |
| 5 | Scores[5] | Điểm của 5 môn học |
| 6 | GPA | Điểm trung bình (GPA) |

### 2.2. Các chức năng

Các hàm chức năng trong hệ thống quản lý sinh viên . (Bảng mô tả các chức năng chính được xây dựng trong chương trình).

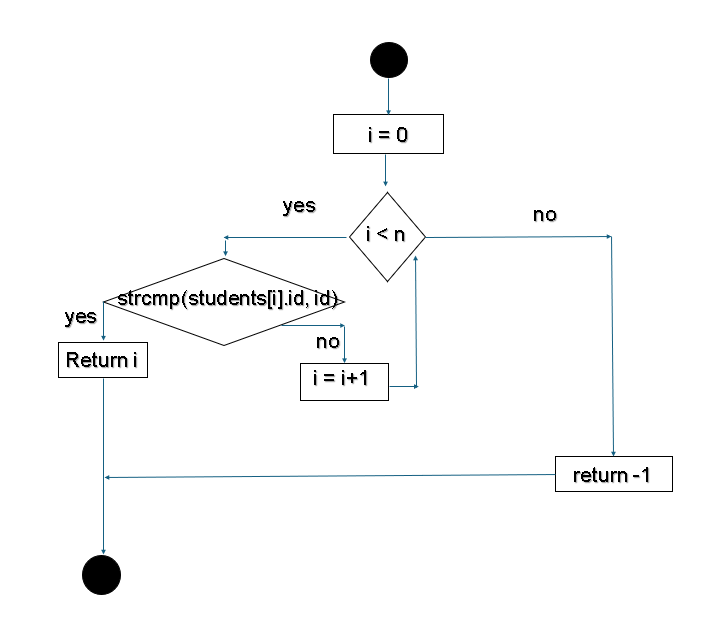
Bảng 3

|  |  |  |
| --- | --- | --- |
| # | Function | Description |
| 1 | Student Management | Quản lý sinh viên |
| 2 | addStudent | Thêm sinh viên mới |
| 3 | editStudent | Sửa thông tin sinh viên |
| 4 | deleteStudent | Xóa sinh viên khỏi danh sách |
| 5 | displayStudents | Hiển thị toàn bộ danh sách sinh viên |
| 6 | searchByID | Tìm sinh viên theo ID |
| 7 | searchByName | Tìm sinh viên theo tên |
| 8 | sortByGPA | Sắp xếp sinh viên theo GPA |
| 9 | sortByName | Sắp xếp sinh viên theo tên (A-Z) |
| 10 | sortByBirthYear | Sắp xếp sinh viên theo năm sinh |
| 11 | calculateStudentGPA | Tính GPA của một sinh viên |
| 12 | calculateOverallGPA | Tính GPA trung bình của toàn bộ sinh viên |
| 13 | saveToFile | Lưu dữ liệu ra file |
| 14 | loadFromFile | Đọc dữ liệu từ file |

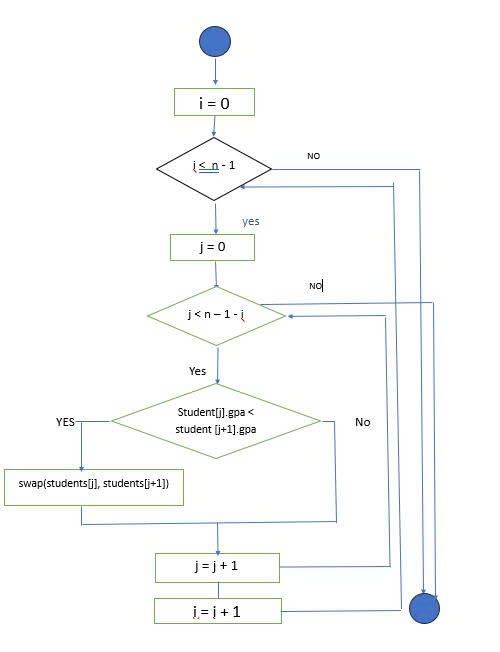
## 3. Thiết kế thuật toán & Sơ đồ khối

Các sơ đồ khối mô tả những chức năng chính như: thêm sinh viên, tính GPA, tìm kiếm, sắp xếp, đọc/ghi file.  
Ví dụ minh họa dưới đây là sơ đồ khối của hàm **“Tìm sinh viên theo mã ID”**.

**SearchByID**

****

**SortByGPA**

****

## 4. Cài đặt các chức năng cơ bản

Chương trình được viết bằng **C** gồm các chức năng chính:

* calculateGPA() (trên struct array)
* Nhập và xuất dữ liệu sinh viên
* Menu điều khiển (dùng switch-case)

**4.1. Chức năng quản lý sinh viên**

* addStudent() – Thêm sinh viên mới (mã, tên, năm sinh, chuyên ngành, điểm 5 môn).
* editStudent() – Sửa thông tin sinh viên qua mã ID.
* deleteStudent() – Xóa sinh viên khỏi danh sách bằng cách dời các phần tử mảng.
* displayStudents() – Hiển thị toàn bộ sinh viên theo dạng bảng (bao gồm GPA và chuyên ngành).

4.2. **Chức năng tìm kiếm và sắp xếp**

* sortByGPA() – Sắp xếp giảm dần theo GPA.
* sortByName() – Sắp xếp theo bảng chữ cái (A–Z).
* sortByBirthYear() – Sắp xếp theo năm sinh (từ già đến trẻ).
* searchByID() – Tìm sinh viên theo mã ID, trả về vị trí.
* searchByName() – Tìm sinh viên theo tên (không phân biệt hoa thường, cho phép tìm một phần tên).

4.3. **Chức năng tính GPA**

* calculateStudentGPA() – Tính GPA cho một sinh viên (trung bình cộng của 5 điểm hợp lệ).
* calculateOverallGPA() – Tính GPA trung bình của tất cả sinh viên trong danh sách.

4.4. **File và Menu**

* saveToFile() và loadFromFile() – Lưu và đọc dữ liệu sinh viên từ file **students.txt**.
* displayMenu() – Hiển thị các lựa chọn: thêm, sửa, xóa, tìm kiếm, sắp xếp, lưu/đọc, tính GPA.
* main() – Điều khiển toàn bộ chương trình bằng **vòng lặp do-while** và **switch-case**.

5. **Thiết kế cấu trúc chương trình**

Chương trình được chia module rõ ràng để dễ bảo trì.

Bảng 4

|  |  |
| --- | --- |
| File Name | Description |
| Student.h | Bao gồm định nghĩa về cấu trúc Sinh viên và khai báo cho tất cả các hàm liên quan đến sinh viên (thêm, chỉnh sửa, xóa, hiển thị, sắp xếp, tìm kiếm và GPA). |
| Student.c | Triển khai tất cả các hàm được khai báo trong student.h, xử lý logic cốt lõi của việc quản lý sinh viên, tính điểm trung bình (GPA), sắp xếp và tìm kiếm. |
| File.h | Khai báo các hàm xử lý tệp (saveToFile() và loadFromFile()). |
| File.c | Triển khai các chức năng đọc và ghi dữ liệu học sinh vào tệp văn bản students.txt. Bao gồm xử lý lỗi cơ bản cho các thao tác trên tệp. |
| Menu.h | Khai báo hàm displayMenu() để hiển thị các tùy chọn cho người dùng. |
| Menu.c | Triển khai hệ thống menu, cho phép người dùng lựa chọn các thao tác như thêm, chỉnh sửa, xóa, sắp xếp, tìm kiếm và lưu/tải dữ liệu. |
| Program.c | Chứa hàm main(). Điều khiển việc thực thi chương trình bằng vòng lặp do-while và cấu trúc switch-case để gọi tất cả các hàm khác. |
| Data\_student.h | Chứa định nghĩa dữ liệu toàn cục, hằng số hoặc khai báo chung được sử dụng bởi nhiều tệp |

## 6. Kiểm tra & Nhập/Xuất tệp

## Hệ thống hỗ trợ lưu và đọc dữ liệu từ tệp văn bản (.txt).

## Xử lý ngoại lệ được áp dụng khi tệp không tồn tại hoặc định dạng không hợp lệ.

## Các trường hợp kiểm tra bao gồm:

## - Thêm học sinh hợp lệ

## - Tìm kiếm học sinh không tồn tại

## - Xử lý lỗi khi mở tệp không thành công

## 6.1. Đọc và ghi file

Hai hàm chính là saveToFile() và loadFromFile() để lưu và đọc dữ liệu theo **định dạng CSV**.  
Đảm bảo dữ liệu được lưu lại giữa các lần chạy chương trình.

**6.2. Xử lý ngoại lệ**

Chương trình bao gồm cách xử lý lỗi tập tin cơ bản:

Hiển thị cảnh báo nếu không thể mở file.

Khởi động với danh sách trống nếu file không tồn tại hoặc rỗng.

Dừng việc đọc dữ liệu một cách an toàn nếu phát hiện lỗi định dạng dữ liệu

6.3 Các trường hợp kiểm thử

Bảng 5

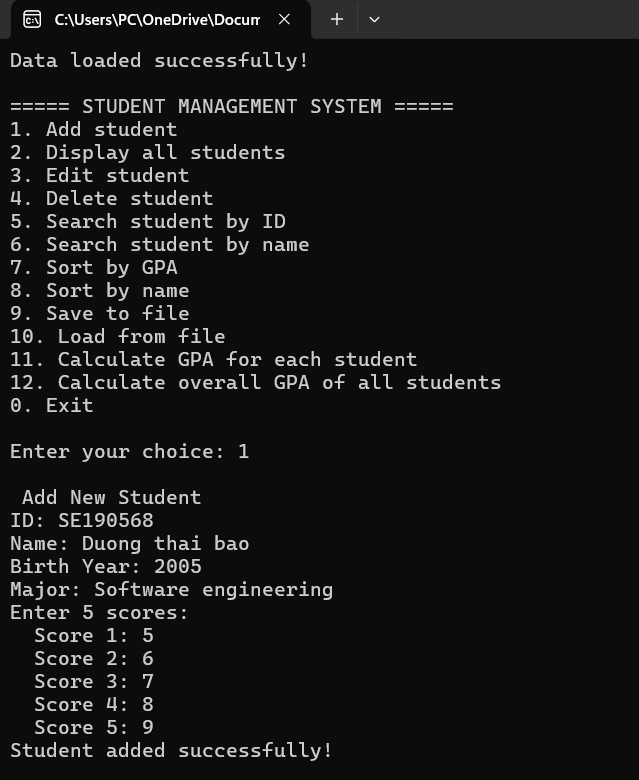
|  |  |  |  |
| --- | --- | --- | --- |
| Test Case | Description | Expected Output | Result |
| 1. Thêm và lưu sinh viên | Thêm bản ghi sinh viên mới rồi chọn “Save to file” | Hiển thị: “Student added successfully!” và “Data saved successfully!” | Passed |
| 1. Tải lại dữ liệu | Khởi động lại chương trình và chọn “Load from file” | Hiển thị: “Data loaded successfully!” và danh sách sinh viên được nạp lên | Passed |
| 3. Tìm sinh viên không tồn tại | Nhập ID không có trong danh sách | Hiển thị: “Not found!” | Passed |
| 4. Xử lý khi thiếu file | Xóa students.txt rồi chạy lại chương trình | Hiển thị: “No file found, starting fresh.”; chương trình chạy với danh sách rỗng | Passed |
| 5. Sửa và lưu lại | Sửa thông tin sinh viên rồi chọn lưu | Nội dung file được cập nhật đúng theo sửa đổi | Passed |

## 7. Kết quả thí nghiệm (Experimental Results)

## Phần này mô tả các kết quả minh họa khi chạy chương trình, kèm chú thích về các màn hình tương tác chính (ảnh chụp màn hình được đề xuất trong báo cáo thực tế).

## 7.1. Thêm sinh viên

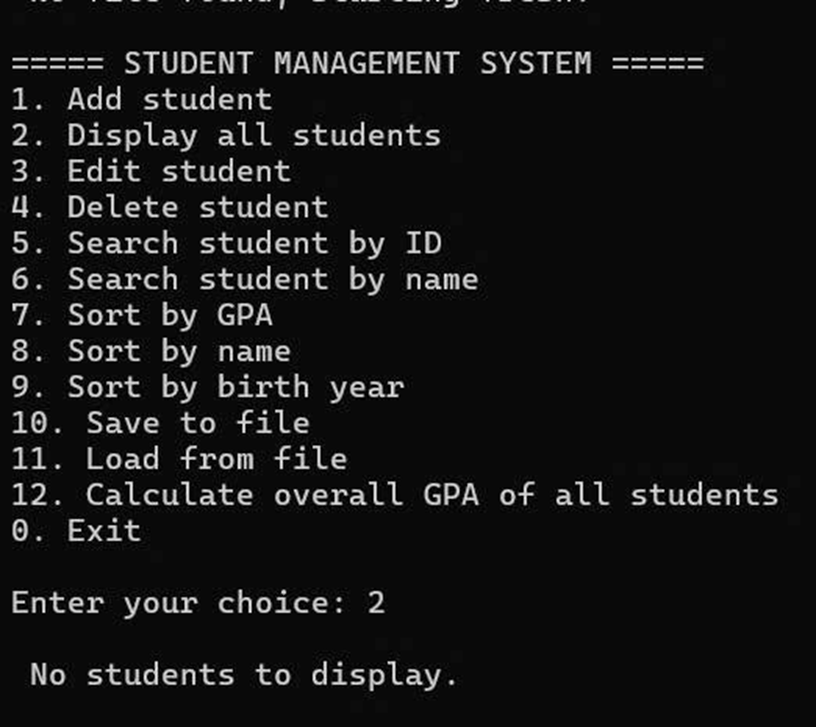
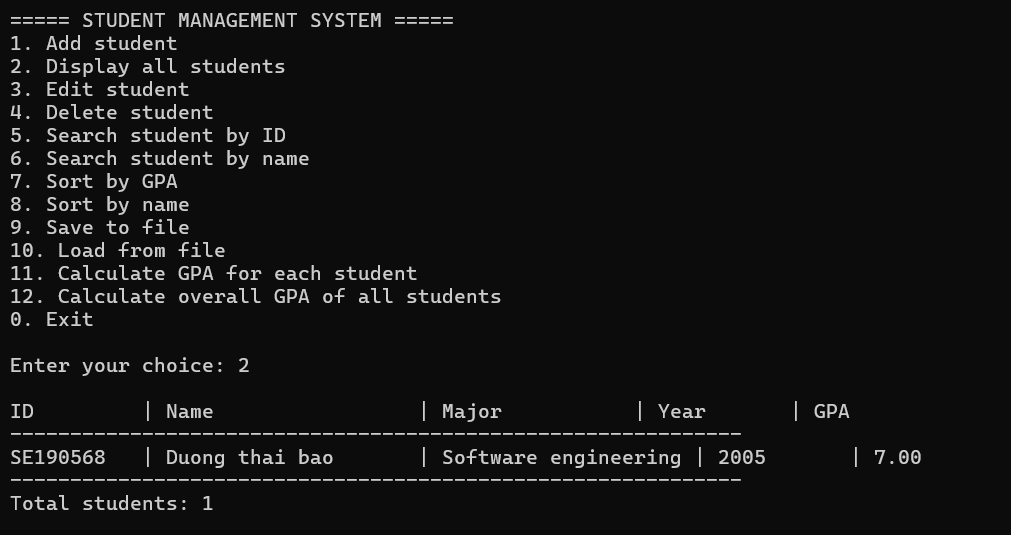
**Khi người dùng chọn chức năng 1. Thêm sinh viên, chương trình sẽ cho phép nhập thông tin sinh viên bao gồm: mã sinh viên, họ tên, năm sinh, chuyên ngành và điểm của 5 môn học.  
Sau khi nhập dữ liệu hợp lệ, chương trình hiển thị thông báo: “Thêm sinh viên thành công!” để xác nhận việc thêm mới.**



7.2 Hiển thị danh sách sinh viên

Khi người dùng chọn chức năng **2. Hiển thị tất cả sinh viên**, chương trình sẽ xuất ra danh sách các sinh viên với định dạng cột rõ ràng, bao gồm: **Mã SV, Họ tên, Chuyên ngành,**

**Năm sinh, Điểm GPA**.

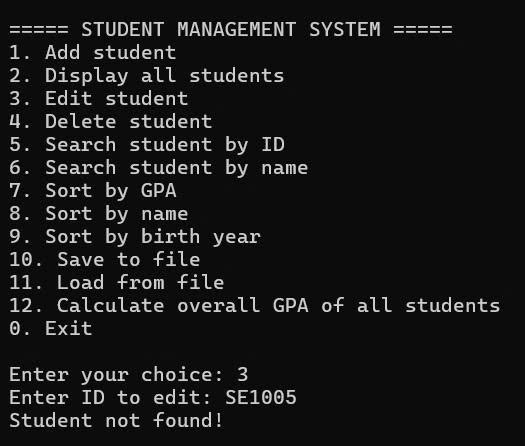
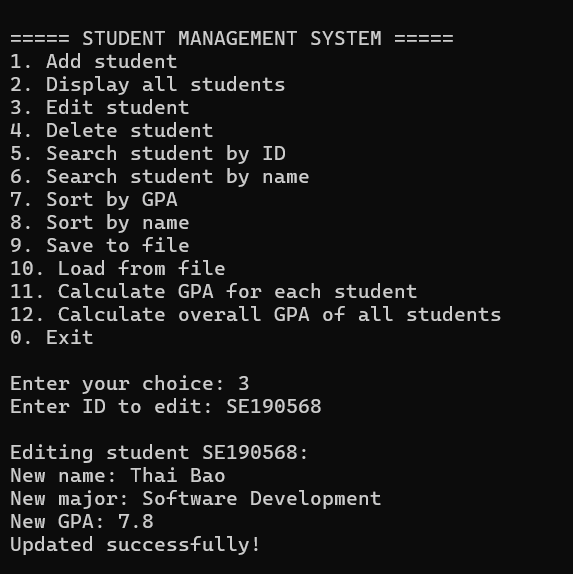


7.3 Chỉnh sửa thông tin sinh viên

Khi người dùng chọn chức năng **3. Chỉnh sửa sinh viên**, chương trình cho phép cập nhật thông tin của sinh viên dựa trên **mã sinh viên**, bao gồm các thông tin như: **họ tên,**

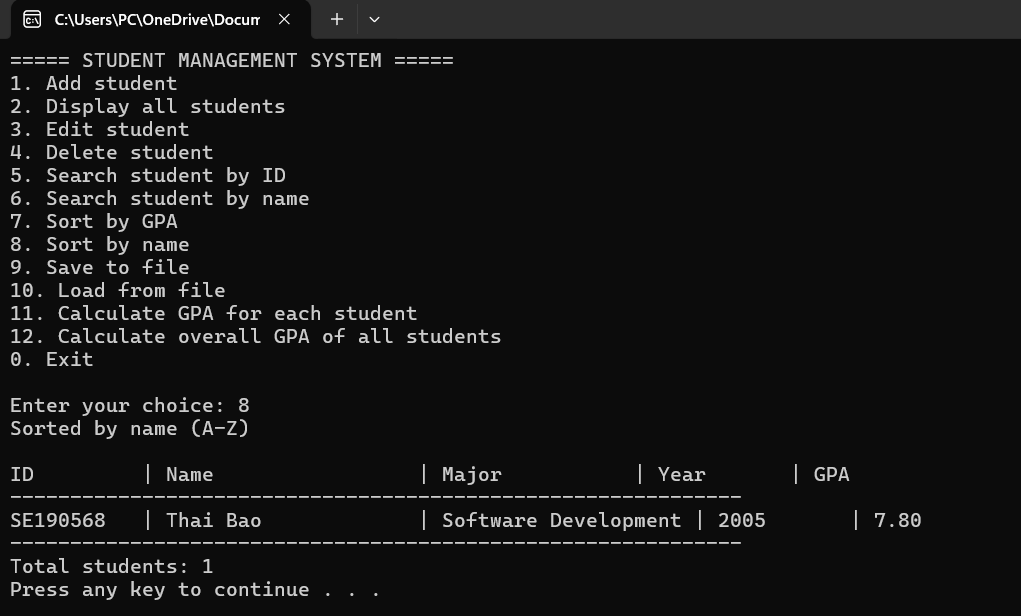
**chuyên ngành hoặc điểm GPA**.

Sau khi chỉnh sửa thành công, chương trình hiển thị thông báo: **“Cập nhật thành công!”**.



7.4 Sắp xếp sinh viên theo tên

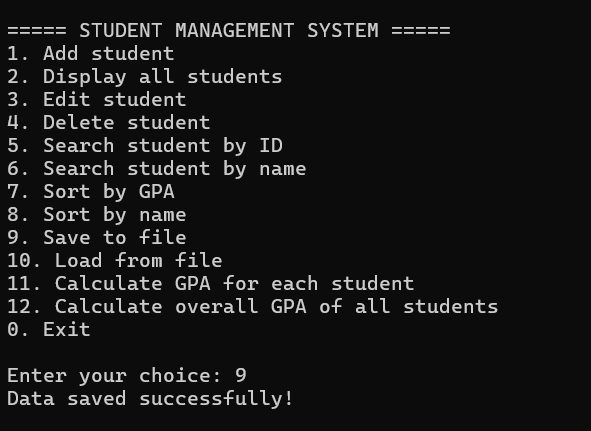
Khi người dùng chọn chức năng **8. Sắp xếp theo tên**, chương trình sẽ sắp xếp danh sách sinh viên theo **thứ tự bảng chữ cái (A → Z)** và hiển thị kết quả đã được sắp xếp.



7.5 Lưu dữ liệu vào file

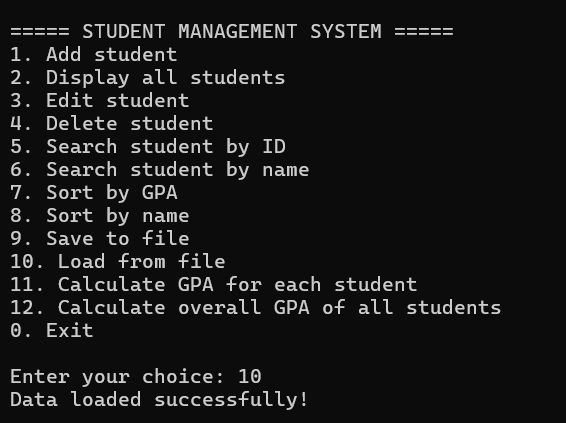
Khi người dùng chọn chức năng **9. Lưu vào file**, chương trình sẽ ghi danh sách sinh viên đã được cập nhật vào file **students.txt**.

Sau khi lưu xong, chương trình hiển thị thông báo: **“Lưu dữ liệu thành công!”**.



7.6 Tải dữ liệu từ file

Khi người dùng chọn chức năng **10. Tải từ file**, chương trình sẽ nạp lại dữ liệu đã lưu từ file vào chương trình và hiển thị thông báo: **“Tải dữ liệu thành công!”**.



## 8. Phụ lục (Appendix)

Danh sách các tệp mã nguồn chính (**Student.c, StudentManager.c, Program.c**) và tệp dữ liệu mẫu (**students.txt**).

8.1.Các tệp mã nguồn( Source Code Files)

Danh sách và mô tả các tệp mã nguồn của chương trình.(Bảng thể hiện cấu trúc các file trong dự án, cùng chức năng và vai trò của từng file.)

Bảng 6

File name Description

|  |  |
| --- | --- |
| Student.h | Chứa định nghĩa cấu trúc Student và các prototype của hàm liên quan. |
| Student.c | Cài đặt các hàm thêm, sửa, xóa, sắp xếp và hiển thị sinh viên. |
| Program.c | Chứa hàm main() và điều khiển luồng hoạt động của chương trình. |
| File.h | Khai báo các hàm liên quan đến xử lý file (nhập/xuất dữ liệu). |
| File.c | Cài đặt các hàm loadFromFile() và saveToFile() để đọc/ghi dữ liệu sinh viên. |
| Menu.h | Khai báo các hàm liên quan đến việc hiển thị menu. |
| Menu.c | Cài đặt giao diện menu chính và xử lý lựa chọn từ người dùng. |
| Data\_student.h | Định nghĩa các hằng số toàn cục hoặc biến dùng chung giữa nhiều module. |

8.2 Tệp dữ liệu(Data Files)

Tệp dữ liệu của hệ thống quản lý sinh viên.(Bảng mô tả tệp dữ liệu được sử dụng để lưu trữ thông tin sinh viên trong chương trình.)

File name Description

|  |  |
| --- | --- |
| students.txt | Lưu trữ thông tin sinh viên như mã số, họ tên, năm sinh, chuyên ngành, GPA và điểm các môn học. |