

ĐẠI HỌC QUỐC GIA THÀNH PHỐ HỒ CHÍ MINH
TRƯỜNG ĐẠI HỌC BÁCH KHOA
KHOA KHOA HỌC - KỸ THUẬT MÁY TÍNH



KỸ THUẬT LẬP TRÌNH (CO1011)

Báo cáo bài tập lớn Chương trình quản lý sinh viên

SV thực hiện: Lê Hữu Vinh - 1614117
Trần Đức Mạnh - 1611992
Trần Văn Hiến - 1611097
Lăng Văn Dương - 1610578

GV hướng dẫn: Trần Tuấn Anh

Tp. Hồ Chí Minh, Tháng 06/2018

Mục lục

1	Phần mở đầu.....	2
1.1	Tổ chức nhóm, đóng góp của các thành viên.....	2
1.2	Mục tiêu.....	3
1.3	Phương pháp nghiên cứu.....	3
1.4	Mô tả các loại người dùng.....	3
1.4.1	Administrator.....	3
1.4.2	Giảng viên.....	4
1.4.3	Sinh viên.....	4
2	Cơ sở lý thuyết.....	4
2.1	Giới thiệu ngôn ngữ.....	4
2.2	Mảng một chiều.....	5
2.2.1	Khái niệm.....	5
2.2.2	Ưu điểm.....	5
2.2.3	Khuyết điểm.....	5
2.2.4	Các thao tác cơ bản.....	5
2.3	Một số kiến thức cơ bản về C/C++.....	6
2.3.1	Kiểu dữ liệu.....	6
2.3.2	Câu lệnh, vòng lặp.....	6
2.3.3	Hàm.....	7
2.3.4	Kiểu dữ liệu struct.....	8
2.3.5	Khai báo thư viện, hằng mặc định.....	8
3	Xây dựng chương trình quản lý sinh viên.....	8
3.1	Sơ đồ các chức năng chính.....	8
3.2	Hàm tìm kiếm theo tên.....	9
3.3	Hàm xóa sinh viên.....	10
3.4	Các modul chính của chương trình.....	11
3.4.1	Đăng nhập.....	11
3.4.2	Administrator.....	11
3.4.3	Giảng viên.....	15
3.4.4	Sinh viên.....	16

1 Phần mở đầu

Ngày nay, tin học đã có những bước tiến nhanh chóng về ứng dụng của nó trong mọi lĩnh vực của cuộc sống trên phạm vi toàn thế giới nói chung và Việt Nam nói riêng. Tin học được người ta quan tâm và nhắc đến hơn bao giờ hết vì nó là một phần không thể thiếu trong cuộc sống văn minh, góp phần đẩy mạnh công cuộc công nghiệp hóa hiện đại hóa đất nước, tiến đến nền kinh tế tri thức. Máy vi tính cùng với những phần mềm là công cụ đắc lực giúp ta quản lý, tổ chức, sắp xếp và xử lý công việc một cách nhanh chóng và chính xác

Ở Việt Nam hiện nay, máy tính điện tử đặc biệt là máy vi tính trong nhiều năm qua đã được sử dụng rất rộng rãi. Sự phát triển của tin học, các công nghệ phần mềm, phần cứng, các tài liệu tham khảo đã đưa chúng ta từng bước tiếp cận với công nghệ thông tin trong mọi lĩnh vực nhằm đáp ứng nhu cầu của con người.

Quản lý sinh viên là một đề tài không còn mới mẻ với các bài toán quản lý. Việc đưa tin học vào ứng dụng để quản lý là rất hữu ích, vì chúng ta phải bỏ ra rất ít thời gian mà lại thu được hiệu quả cao, rất chính xác và tiện lợi nhanh chóng.

Với khoảng thời gian không nhiều, vừa phân tích thiết kế, nghiên cứu tìm hiểu khai thác ngôn ngữ mới, vừa thực hiện chương trình quả là khó khăn đối với chúng em. Xong cùng với sự nỗ lực của nhóm, nhóm em đã hoàn thành bài tập của mình theo đúng thời gian quy định. Tuy nhiên bài vẫn còn nhiều thiếu sót xót mong thầy châm chước cho nhóm em, chúng em xin chân thành cảm ơn.

1.1 Tổ chức nhóm, đóng góp của các thành viên

ST T	Họ Và Tên	MSSV	% đóng góp
1	Lê Hữu Vinh	16141	25

		17	
2	Trần Đức Mạnh	1611992	25
3	Trần Văn Hiến	1611097	30
4	Lăng Văn Dương	1610578	20

1.2 Mục tiêu

- Xây dựng chương trình quản lý sinh viên bằng ngôn ngữ C/C++ đơn giản, dễ sử dụng. Giải quyết các vấn đề: Đăng nhập, đăng xuất, đổi mật khẩu, thêm-xóa giảng viên, sinh viên, mở môn học, tổng kết điểm, sửa điểm, tìm kiếm theo tên, xóa tên,...
- Giúp hiểu rõ hơn về các khái niệm cơ bản trong C/C++: struct, class, cách đọc, ghi file và nâng cao khả năng lập trình.

1.3 Phương pháp nghiên cứu

- Ứng dụng các kiến thức đã học, đọc tham khảo một số tài liệu về lập trình C/C++.
- Tham khảo, quan sát các phần mềm quản lý sinh viên trong thực tế, trên cơ sở đó tiến hành phân tích thiết kế chương trình quản lý sinh viên.

1.4 Mô tả các loại người dùng

Có 3 loại người dùng: *Administrator*, *giảng viên* và *sinh viên*.

Mọi tài khoản đều có thể đăng xuất, đổi mật khẩu và khi đăng nhập sai có thể đăng nhập lại.

1.4.1 Administrator

- Có thể thêm - xóa giảng viên và sinh viên, tìm kiếm các user trong hệ thống. Khi tạo user admin tạo mật khẩu giống với tên user.
- Tìm kiếm theo tên: Administrator có thể tìm được nhiều user có cùng keyword theo tên.
- Xóa theo tên: Khi nhiều sinh viên tốt nghiệp có administrator có thể xóa nhiều sinh viên một lúc.

1.4.2 Giảng viên

- Mở môn học: Mỗi giảng viên có thể mở tối đa 5 môn học trong một học kì.
- Tổng kết điểm của sinh viên với mỗi môn đã mở.
- Sửa điểm của sinh viên với mỗi môn đã mở: Hệ thống sẽ lưu lại mỗi lần sửa điểm.

1.4.3 Sinh viên

- Đăng kí môn học - hủy môn học: Mỗi sinh viên có thể đăng kí tối đa 20 tín chỉ trong 1 học kì.
- Xem thông tin cá nhân: Xem các môn đã đăng kí, các môn đang học.
- Tìm kiếm theo tên: Sinh viên có thể tìm được nhiều môn học có cùng keyword theo tên.

2 Cơ sở lý thuyết

2.1 Giới thiệu ngôn ngữ

C là ngôn ngữ lập trình cấp cao, được sử dụng rất phổ biến để lập trình hệ thống cùng với Assembler và phát triển các ứng dụng.

Ngôn ngữ lập trình C là một ngôn ngữ lập trình hệ thống rất mạnh và rất “mềm dẻo”, có một thư viện gồm rất nhiều các hàm (function) đã được tạo sẵn. Người lập trình có thể tận dụng các hàm này để giải quyết các bài toán mà không cần phải tạo mới. Hơn thế nữa, ngôn ngữ C hỗ trợ rất nhiều phép toán nên phù hợp cho việc giải quyết các bài toán kỹ thuật có nhiều công thức phức tạp. Ngoài ra, C cũng cho phép người lập trình tự định nghĩa thêm các kiểu dữ liệu trừu tượng khác. Tuy nhiên, điều mà người mới vừa học lập trình C thường gặp “rắc rối” là “hơi khó hiểu” do sự “mềm dẻo” của C. Dù vậy, C được phổ biến khá rộng rãi và đã trở thành một công cụ lập trình khá mạnh, được sử dụng như là một ngôn ngữ lập trình chủ yếu trong việc xây dựng những phần mềm hiện nay.

2.2 Mảng một chiều

2.2.1 Khái niệm

Mảng là kiểu dữ liệu có cấu trúc bao gồm nhiều phần tử kiểu và được bố trí vùng nhớ liên tục.

Kiểu của các phần tử mảng gọi là kiểu cơ sở. Mỗi phần tử mảng là một biến có kiểu cơ sở.

Mảng có kích thước là số phần tử trong mảng. Kích thước mảng bắt buộc phải là biểu thức hằng nguyên để có thể cấp phát vùng nhớ lúc biên dịch.

2.2.2 Ưu điểm

Ưu điểm của mảng là giúp ta tiết kiệm tài nguyên hệ thống. Ta có thể sử dụng một mảng có kích thước lớn trong một thời gian nào đó rồi xóa bỏ để trả lại vùng nhớ cho hệ thống. Dễ cài đặt và truy nhập các phần tử dữ liệu. Tốc độ truy nhập đến một vị trí bất kỳ trên mảng nhanh, hiệu quả.

2.2.3 Khuyết điểm

Cần phải xác định trước số phần tử mảng trước khi sử dụng => không phù hợp với các bài toán chưa biết trước số lượng phần tử.

Khó khăn trong các thao tác chèn và xóa một phần tử bất kỳ trong mảng.

Nếu bài toán mà việc chèn phần tử xóa phần tử diễn ra liên tục thì tốc độ xử lý sẽ rất chậm.

2.2.4 Các thao tác cơ bản

- Nhập giá trị cho các phần tử mảng.
- Xuất giá trị các phần tử mảng (ra màn hình).
- Thêm 1 phần tử vào mảng.
- Xóa một phần tử ra khỏi mảng.
- Tìm kiếm trên mảng.
- Sắp xếp mảng.

❖ Các thao tác trên thực chất là duyệt mảng và xử lý trên từng phần tử mảng nhờ vào cấu trúc lặp một cách tổng quát như sau:

+ Hàm nhập giá trị cho các phần tử mảng từ bàn phím:

```
void Input(int a[], int n) {  
    for (int i=0; i<n; i++) {  
        cout << "a[" << i << "] = "; cin >> a[i];  
    }  
}
```

```
}  
+ Hàm xuất giá trị cho các phần tử mảng 1 chiều ra màn hình  
void Output(const int a[], int n) {  
    for (int i=0; i<n; i++) cout << setw(4) <<a [i]; cout <<  
endl;  
}
```

2.3 Một số kiến thức cơ bản về C/C++

2.3.1 Kiểu dữ liệu

string	kiểu ký tự
int	kiểu số nguyên
float	kiểu số thực

2.3.2 Câu lệnh, vòng lặp

Lệnh **if**:

```
if(biểu thức)  
    [lệnh];
```

Lệnh **switch case**:

```
switch(biểu_thức_nguyên)  
{  
    case hằng_1: [lệnh_1;]  
    case hằng_2: [lệnh_2;]  
    case hằng_n: [lệnh_n;]  
    default: [lệnh_n+1;]  
}
```

Lệnh **break**: lệnh break dùng để kết thúc vòng lặp khi chương trình gặp lệnh này thì chương trình sẽ dừng ngay lập tức vòng lặp trong nhất chứa nó.

Vòng lặp **for**:

```
for([bt1];[bt2];[bt3])
```



```
        [lệnh];  
Vòng lặp while:  
        while(biểu thức)  
            [lệnh];  
Vòng lặp do...while:  
        do{  
            [lệnh];  
        }while(bt);
```

2.3.3 Hàm

```
[kiểu trả về/void]<tên  
hàm([danh_sách_tham_số_hình_thức])>  
    <{[khai báo biến cục bộ]  
        [các câu lệnh]  
        [return giá_trị_trả_về;]  
        [các câu lệnh]  
        [return giá_trị_trả_về;]  
    }>
```

2.3.4 Kiểu dữ liệu struct

```
struct kiểu_struct{  
        kiểu_đã_có_1: danh sách các trường cùng  
kiểu;  
        kiểu_đã_có_2: danh sách các trường cùng  
kiểu;  
        kiểu_đã_có_n: danh sách các trường cùng  
kiểu;  
};
```

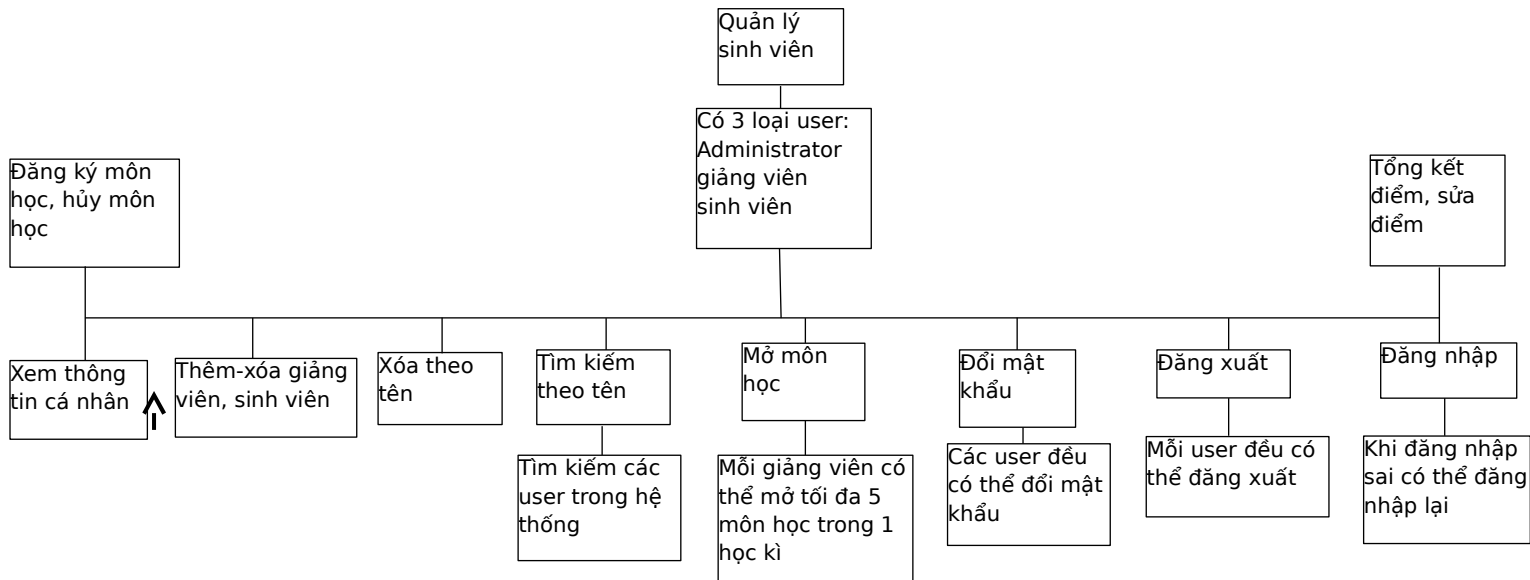
2.3.5 Khai báo thư viện, hằng mặc định

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

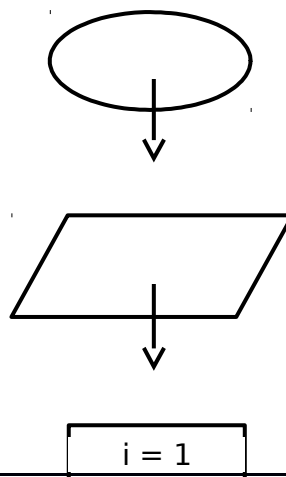
```
using namespace std;
```

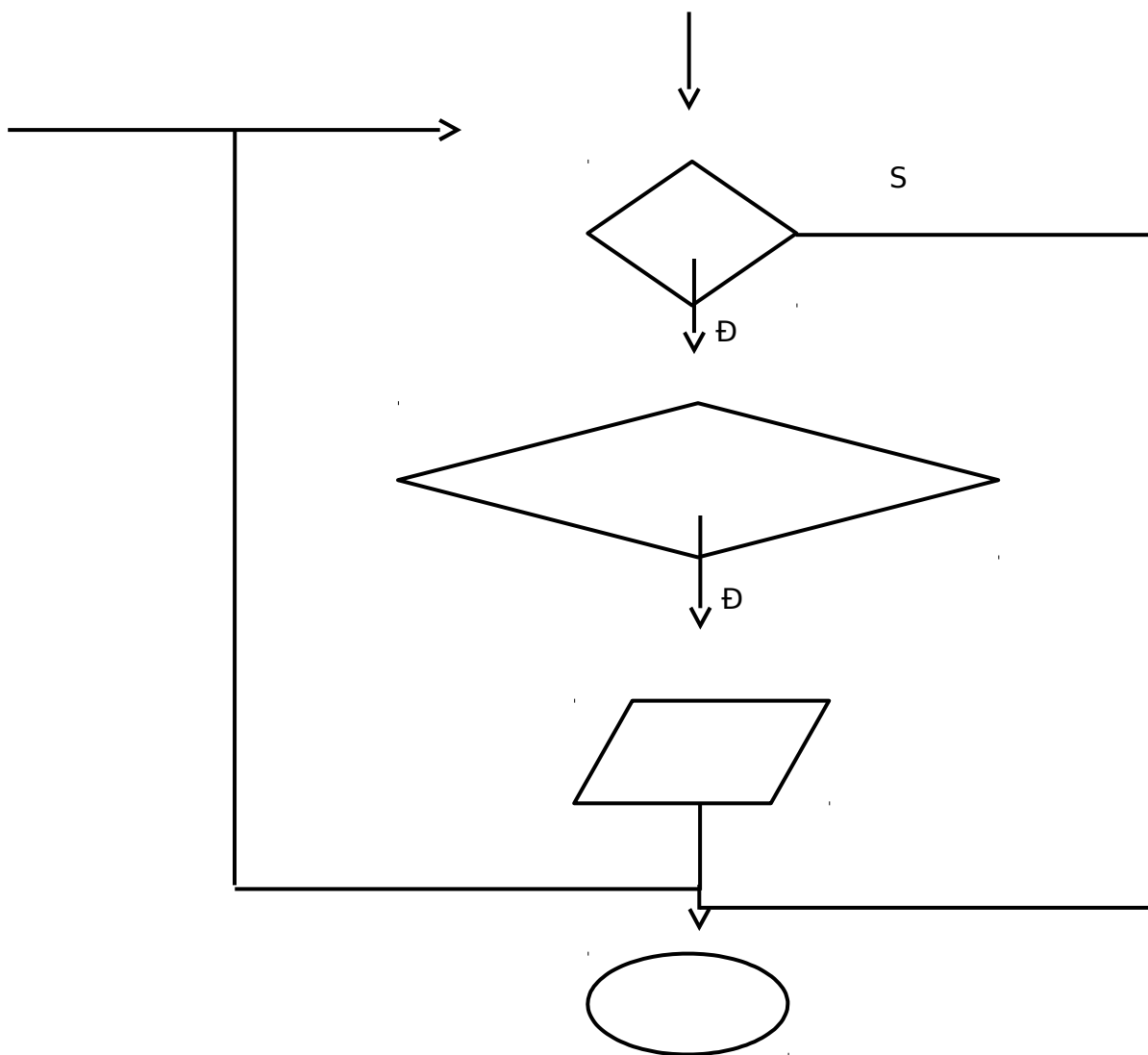
3 Xây dựng chương trình quản lý sinh viên

3.1 Sơ đồ các chức năng chính

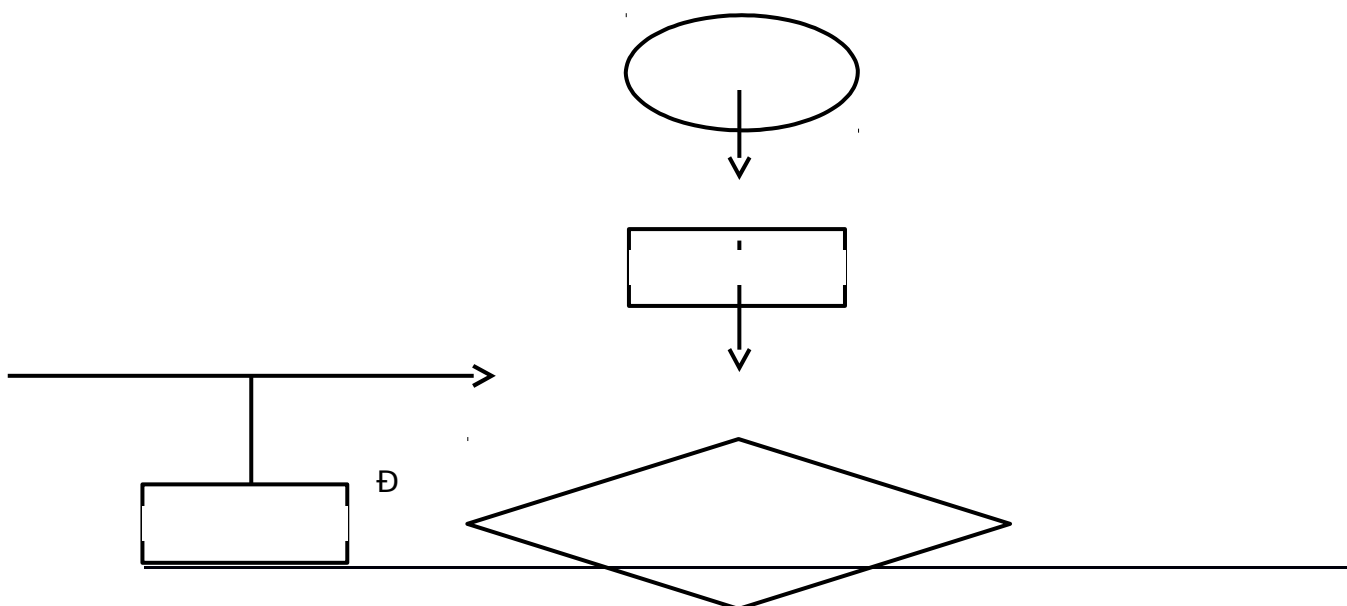


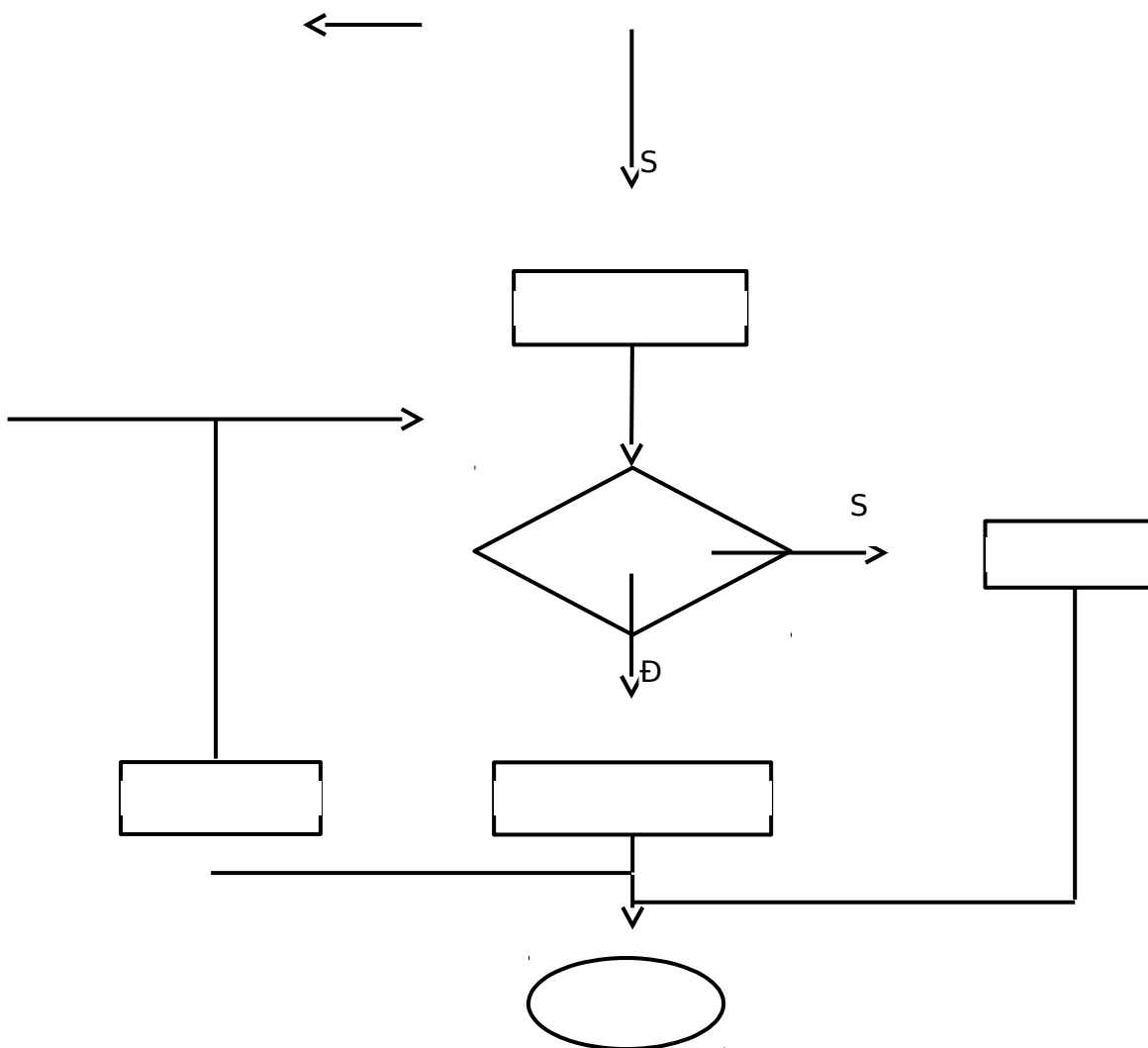
3.2 Hàm tìm kiếm theo tên





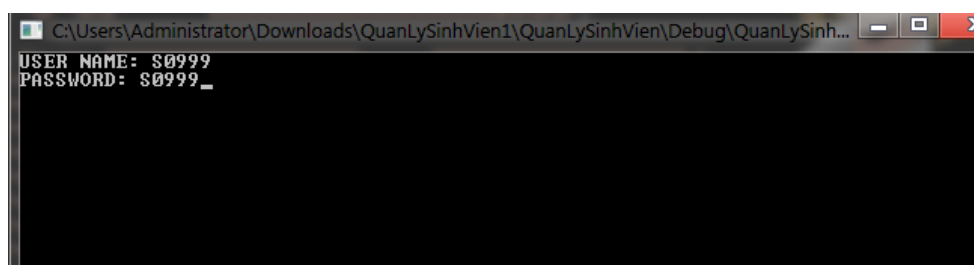
3.3 Hàm xóa sinh viên





3.4 Các modul chính của chương trình

3.4.1 Đăng nhập



- Nhập *tài khoản* và *mật khẩu*

3.4.2 Administrator

3.4.2.1 Giao diện khi đăng nhập

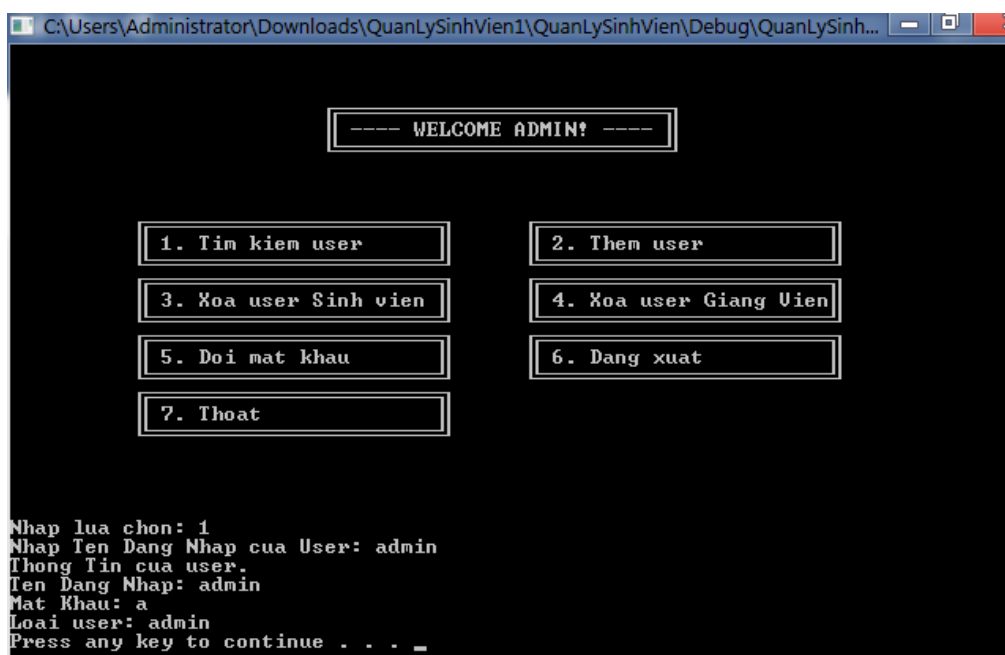


Có các chức năng cho admin như: tìm kiếm user, thêm user, xóa giảng viên – sinh viên, đổi mật khẩu, đăng xuất để đăng nhập tài khoản khác hoặc thoát luôn khỏi chương trình.

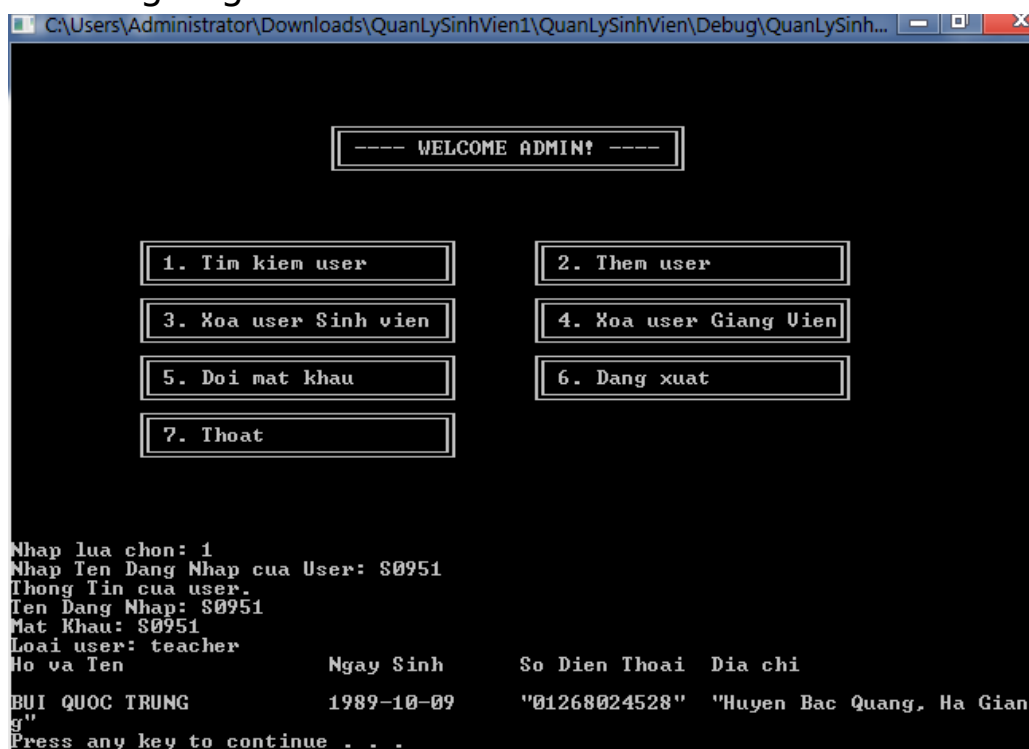
3.4.2.2 Tìm kiếm user

Nhập tên đăng nhập của user và xem thông tin của user đó.

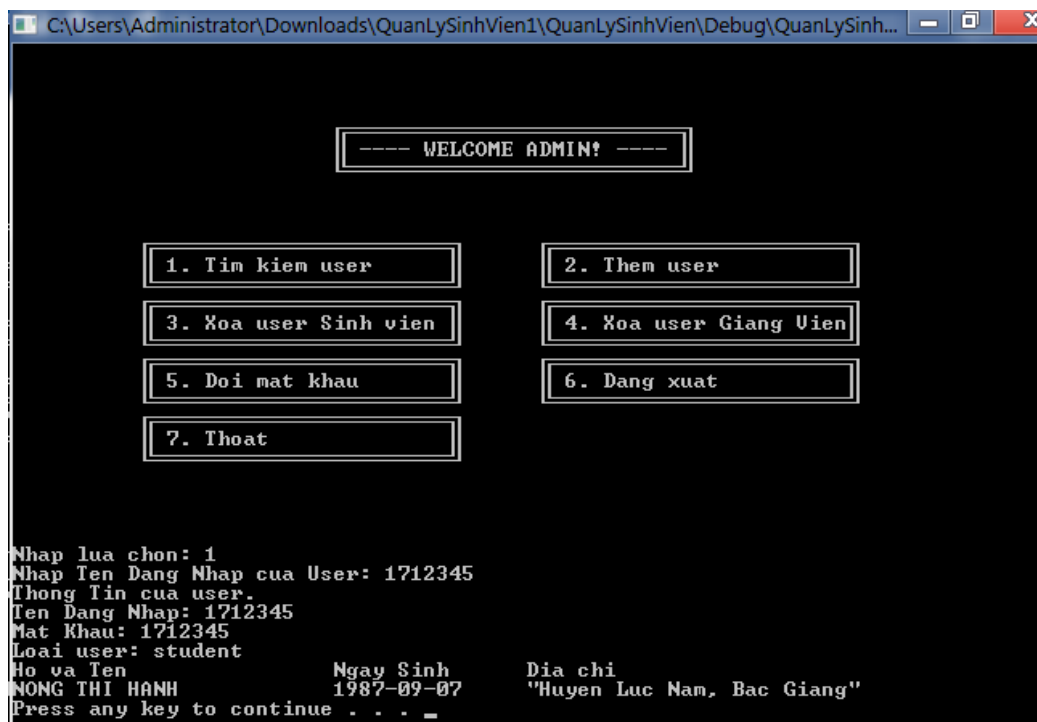
VD: user là admin



VD: user là giảng viên



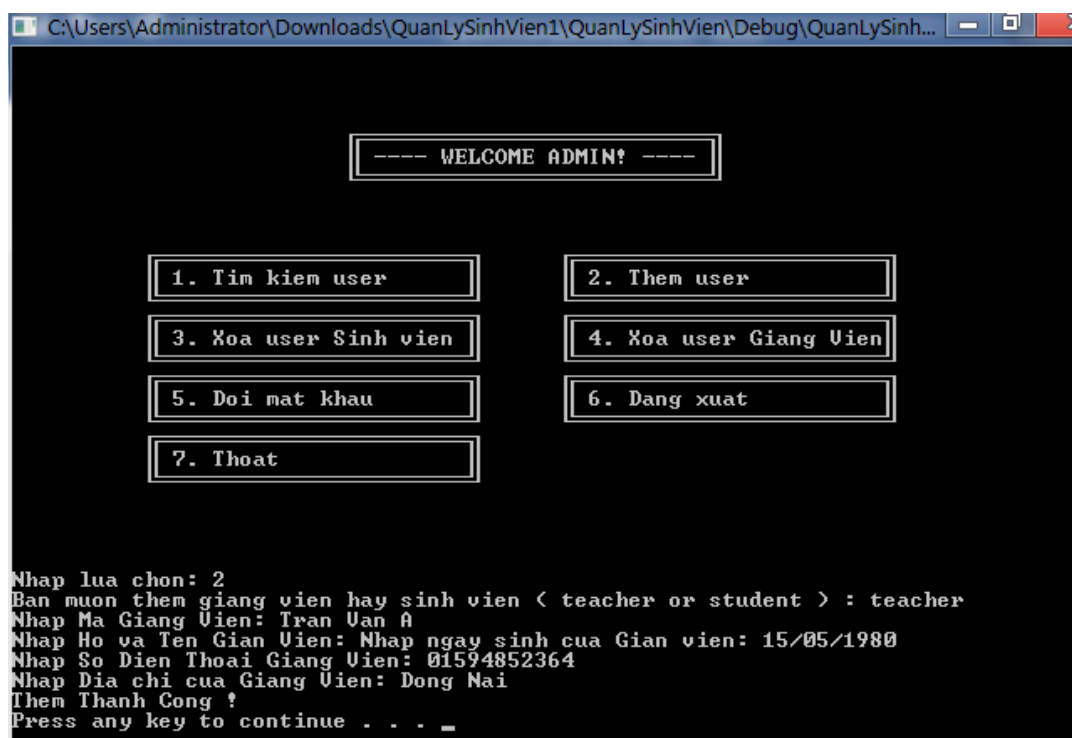
VD: user là sinh viên



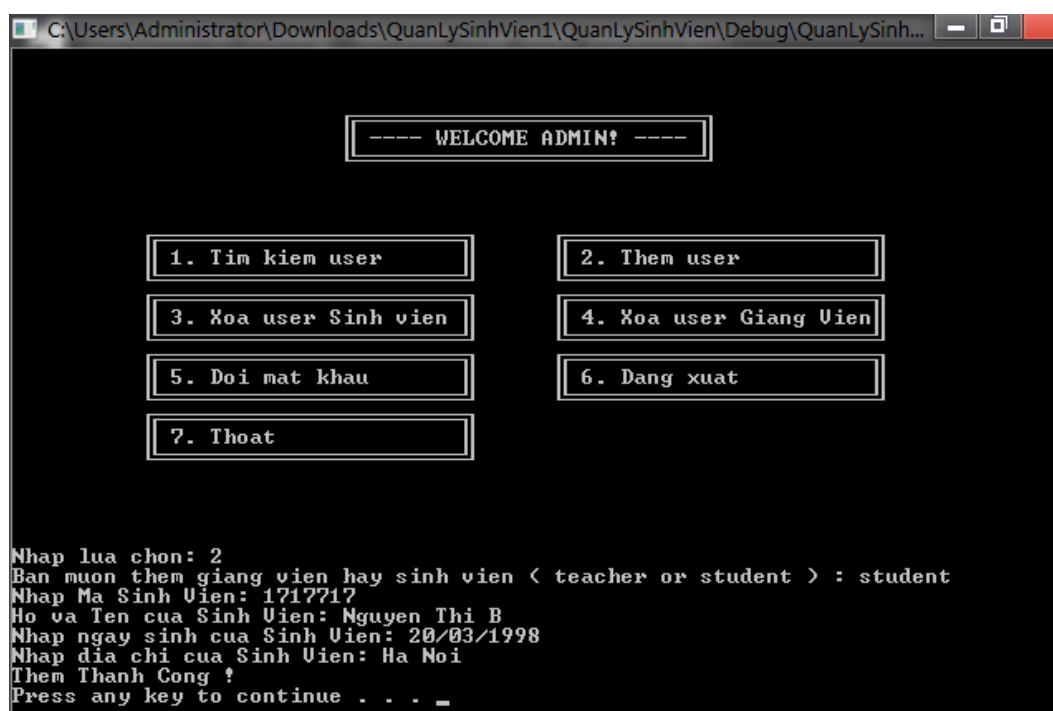
3.4.2.3 Thêm user

Ta lựa chọn thêm giảng viên hoặc sinh viên: thêm giảng viên thì gõ teacher, thêm sinh viên thì gõ student. Sau đó nhập các thông tin của user tương ứng.

VD thêm giảng viên



VD thêm sinh viên



3.4.2.4 Xóa user sinh viên và giảng viên

Ta tìm kiếm user theo tên và sau đó nhập vào mã của user ta muốn xóa.

3.4.2.5 Đổi mật khẩu



- Ta nhập vào mật khẩu cũ (mật khẩu hiện tại) và mật khẩu mới ta cần đổi.

3.4.2.6 Đăng xuất hoặc thoát

- Chọn đăng xuất nếu muốn đăng nhập tài khoản khác hoặc thoát ra.

3.4.3 Giảng viên

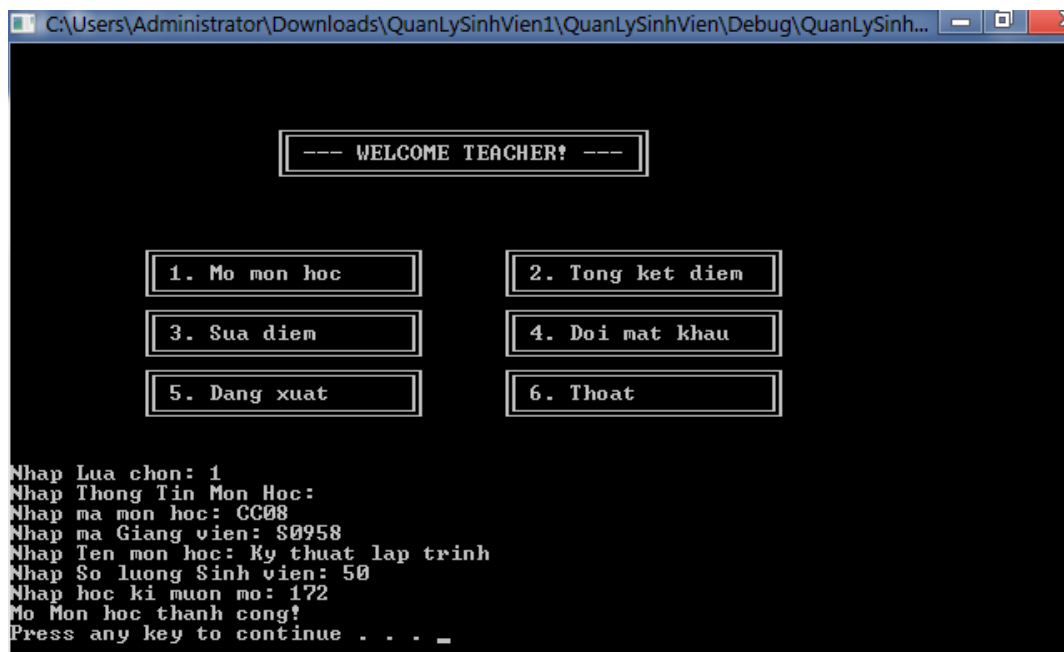
3.4.3.1 Giao diện khi đăng nhập



- Có 6 lựa chọn là: mở môn học, tổng kết điểm, sửa điểm, đổi mật khẩu, đăng xuất và thoát. Ta nhập vào các số nguyên tương ứng trên giao diện.

3.4.3.2 Mở môn học

Nhập vào thông tin của môn học ta định mở: mã môn học, nhập mã giảng viên, tên môn học, số lượng sinh viên, học kì muốn mở.



3.4.4 Sinh viên

3.4.4.1 Giao diện



- Sinh viên có thể: đăng ký môn học, hủy môn học, xem thông tin cá nhân, đổi mật khẩu, đăng xuất và thoát.

3.4.4.2 Đăng ký môn học

Nhập vào tên của môn học muốn đăng ký, ID môn học muốn đăng ký, ID của giảng viên dạy môn đó.

```

ID mon hoc  ID giao vien  Ten mon hoc          So luong  Slot  Hoc ky
-----
Course_ID  Teacher_ID  Name                  n_student slot
CC01S1951,Mo Hinh Hoa,1,50
CC01        S1952      Mo Hinh Hoa           0         30
CC02 S1952,He Thong Nhung,1,50
CC03        S1953      Lap Trinh Web         1         50
CC04 S1954,He Thong Thong Minh,1,50
CC05        S1955      He quan tri CSDL      1         50
CC06 S1956,He dieu hanh,1,50
CC07        S1957      Toan roi rac 2        1         50
CC08 S1958,LT Android,1,50
CC08        S0958      LT Android            0         50      171
CC09        S0958      afd                   0         20      171
CC05        S0951      a                     1         30      181
C01011     S0951      Ky thuat lap trinh    0         50      171

moi ban nhap id mon hoc: CC03
moi ban nhap id giao vien: S1953
dang ki thanh cong!!
Press any key to continue . . .
  
```

3.4.4.3 Hủy môn học

Nhập ID của môn học và ID của giáo viên của môn học mình đã đăng ký.



3.4.4.4 Xem thông tin cá nhân

Xem các môn học mà mình đã đăng ký.