

KHOA ĐÀO TẠO CHẤT LƯỢNG CAO NGÀNH CÔNG NGHỆ THÔNG TIN

BÁO CÁO PROJECT

Quản Lý Học Viên Bằng Cây Nhị Phân Tìm Kiếm

Giáo viên hướng dẫn: Nguyễn Quang Ngọc

Tên sinh viên 1: Huỳnh Thanh Tuấn

MSSV: 20110120

Tên sinh viên 2: Nguyễn Thành Đạt

MSSV: 20110121

Tên sinh viên 3: Nguyễn Hồng Vũ Khánh

MSSV: 19151241

MÔ TẢ ĐỀ TÀI:

- 1) Nhập một lớp học ở dạng cây nhị phân tìm kiếm (BST), mỗi học viên có các thông tin: mã số học viên (MSHV, chuỗi 8 ký tự không dấu, không khoảng trắng) được xem là key, họ tên (có thể có tên lót), (HTHV, chuỗi 30 ký tự không dấu), mã môn học 1 (MMH1, chuỗi 4 ký tự không dấu), điểm môn 1 (DM1, số nguyên), mã môn học 2 (MMH2, chuỗi 4 ký tự không dấu), điểm môn 2 (DM2, số nguyên).
- 2) Thêm 1 học viên mới. (MSHV, HTHV, MMH1, DM1, MMH2, DM2) vào BST.
- 3) In ra màn hình HTHV, DM1, DM2 khi biết MSHV.
- 4) Xóa 1 học viên khi biết MSHV. (Thông báo đã xóa / không có hv trong BST)
- 5) Viết ra màn hình: MSHV, HTHV, DM1 của các học viên theo thứ tự tăng dần của DM1.
- 6) Viết ra màn hình: MSHV, HTHV, DM2 của các học viên theo thứ tự tăng dần của DM2.
- 7) Viết ra màn hình: MSHV, HTHV, DM1 của các học viên theo thứ tự tăng dần của DM1, nếu DM1 bằng nhau thì sắp xếp tăng dần theo DM2.
- 8) In ra màn hình HTHV, DM1, của các học viên theo MMH1 (Tương tự: MMH2, MMHP1, MMHP2, ...).
- 9) Viết ra màn hình MSHV, HTHV dạng cây đang lưu trữ.
- 10) Ghi thông tin vào file (lưu trữ dữ liệu, cho phép đánh đường dẫn chứa file)
- 11) Mở file (đã có để sử dụng) (cho phép đánh đường dẫn chứa file, người dùng có thể thực hiện các yêu cầu 3)-10) trên dữ liệu của file).

CẤU TRÚC DỮ LIỆU ĐƯỢC SỬ DỤNG:

Cây nhị phân tìm kiếm (BST):

+ Khai báo: struct NODE

+ Ý nghĩa: Dùng để chứa thông tin các học viên trong lớp học bao gồm: mã số học viên, họ tên, mã môn học 1, điểm môn 1, mã môn học 2, điểm môn 2 và danh sách các môn học phụ của mỗi học viên (nếu có) trong đó mỗi môn học phụ gồm có: mã môn học và điểm môn học.

CHƯƠNG TRÌNH CON:

- 1) Khai báo: int Xoa_vung_nho_tam()

Chức năng: Xóa các ký tự được nhập từ bàn phím còn lưu trữ trong vùng nhớ tạm tại thời điểm hiện tại đến khi gặp ký tự \n và xóa cả ký tự \n.

- 2) Khai báo: void Nhap_diem_so(int &kq)

Chức năng: Nhập giá trị cho kq là một số nguyên nằm trong phạm vi từ 0 đến 10.

- 3) Khai báo: int KT_khoang_trang(char A[])

- Chức năng: Kiểm tra trong chuỗi A có khoảng trắng hay không.
- 4) Khai báo: void Nhapdethoat()
Chức năng: Dừng để dừng lại xem kết quả.
- 5) Khai báo: int Xoa_Xuong_Dong(char A[])
Chức năng: Dừng để xóa ký tự \n trước ký tự kết thúc chuỗi trong chuỗi A nếu có.
- 6) Khai báo: int Duyet_KT_list(List *T, char A[])
Chức năng: Kiểm tra trong T đã có chuỗi A hay không.
- 7) Khai báo: NODE *Tim_kiem(NODE *T, char MS[])
Chức năng: tìm nút trong cây nhị phân tìm kiếm T có chứa chuỗi MS.
- 8) Khai báo: void Duyet_LNR_1(NODE *x)
Chức năng: Duyệt cây theo thứ tự giữa và in ra mã số học viên, họ tên học viên và điểm môn 1 của các nút.
- 9) Khai báo: void Duyet_LNR_2(NODE *x)
Chức năng: Duyệt cây theo thứ tự giữa và in ra mã số học viên, họ tên học viên và điểm môn 2 của các nút.
- 10) Khai báo: void Duyet_LNR_3(NODE *x)
Chức năng: Duyệt cây theo thứ tự giữa và in ra mã số học viên, họ tên học viên, điểm môn 1 và điểm môn 2 của các nút.
- 11) Khai báo: void AddSV(NODE *&T, HocVien X)
Chức năng: Thêm X vào cây nhị phân tìm kiếm T với nút của cây con bên trái sẽ có Maso nhỏ hơn Maso của nút cha, nút của cây con bên phải sẽ có Maso lớn hơn Maso của nút cha.
- 12) Khai báo: void Them_vao_List(NODE *T):
Chức năng: Thêm mã và điểm môn học phụ cho học viên.
- 13) Khai báo: void NhapSV(HocVien &A)
Chức năng: Nhập thông tin cho một học viên.
- 14) Khai báo: void Them_hoc_vien(NODE *&T)
Chức năng: kiểm tra học viên thêm vào, nếu mã số học viên thêm vào không đúng 8 ký tự hoặc có dấu khoảng trắng hoặc đã có trong danh sách thì yêu cầu nhập lại mã số học viên, nếu mã số học viên nhập vào đúng và khác mã số học viên của các nút trong T thì cho phép nhập các thông tin còn lại và thực hiện thêm học viên vào cây nhị phân tìm kiếm.
- 15) Khai báo: void Them_HP(NODE *T)
Chức năng: thực hiện nhập mã số học viên, nếu mã số học viên không đúng 8 ký tự hoặc có dấu khoảng trắng thì nhập lại. Khi nhập đúng thì kiểm tra mã số học viên đó có trong danh sách không, nếu có thì thêm môn học phụ cho học viên còn không có thì in ra không có học viên trong BST.
- 16) Khai báo: void Inthongtin_theoMHV(NODE *T)
Chức năng: kiểm tra mã số học viên nhập vào nếu không đúng 8 ký tự hoặc có dấu khoảng trắng thì nhập lại, nếu đúng thì tìm kiếm mã số học viên đó. Nếu có mã số học viên đó trong danh sách thì in ra thông tin họ tên, điểm môn 1, điểm môn 2. Nếu không có mã số học viên đó trong danh sách thì in ra màn hình không tìm thấy mã số học viên.
- 17) Khai báo: NODE *Trai_nhat(NODE *x)

Chức năng: trả về nút trái nhất của cây con bên trái của nút x. Nếu không có thì trả về nút x.

18) Khai báo: `NODE *Xoa_node(NODE *&ROOT, NODE *T)`

Chức năng: thực hiện xóa nút T trong cây nhị phân tìm kiếm ROOT.

19) Khai báo: `void Goi_xoa(NODE *&T)`

Chức năng: kiểm tra mã số học viên nhập vào nếu không đúng 8 kí tự hoặc có dấu khoảng trắng thì nhập lại, nếu đúng thì tìm kiếm mã số học viên đó. Nếu có mã số học viên đó trong danh sách thì thực hiện xóa học viên đó và sau khi xóa in ra màn hình đã xóa. Nếu không có mã số học viên đó trong danh sách thì in ra màn hình không có học viên trong BST.

20) Khai báo: `void AddSVMH1(NODE *&T, HocVien X)`

Chức năng: thêm X vào cây nhị phân tìm kiếm T với nút của cây con bên trái sẽ có điểm môn 1 nhỏ hơn điểm môn 1 của nút cha, nút của cây con bên phải sẽ có điểm môn 1 lớn hơn hoặc bằng điểm môn 1 nút cha.

21) Khai báo: `void Duyet_NLR_1(NODE *x, NODE *&T1)`

Chức năng: thêm học viên vào cây nhị phân tìm kiếm T1 bằng cách gọi chương trình con: `void AddSVMH1` và duyệt theo thứ tự trước của cây nhị phân tìm kiếm x.

22) Khai báo: `void Tang_dan_DM1(NODE *T)`

Chức năng: Khởi tạo giá trị NULL ban đầu cho cây nhị phân tìm kiếm T1 và gọi các chương trình con: `Duyet_NLR_1`, `Duyet_LNR_1`. In ra màn hình tên các cột: Ma so hoc vien, Ho ten hoc vien, Diem mon 1.

23) Khai báo: `void AddSVMH2(NODE *&T, HocVien X)`

Chức năng: thêm X vào cây nhị phân tìm kiếm T với nút của cây con bên trái sẽ có điểm môn 2 nhỏ hơn điểm môn 2 của nút cha, nút của cây con bên phải sẽ có điểm môn 2 lớn hơn hoặc bằng điểm môn 2 nút cha.

24) Khai báo: `void Duyet_NLR_2(NODE *x, NODE *&T1)`

Chức năng: thêm học viên vào cây nhị phân tìm kiếm T1 theo điểm môn 2 bằng cách gọi chương trình con: `AddSVMH2` và duyệt theo thứ tự trước của cây nhị phân tìm kiếm x.

25) Khai báo: `void Tang_dan_DM2(NODE *T)`

Chức năng: Khởi tạo giá trị NULL ban đầu cho cây nhị phân tìm kiếm T1 và gọi các chương trình con `Duyet_NLR_2`, `Duyet_LNR_2`. In ra màn hình tên các cột: Ma so hoc vien, Ho ten hoc vien, Diem mon 2.

26) Khai báo: `int sosanh(NODE *x, NODE *y)`

Chức năng: dùng để so sánh DM1 của x và y. Nếu DM1 của x và y bằng nhau thì so sánh DM2. Trả về 1 nếu DM1 của x nhỏ hơn DM1 của y hoặc DM2 của x nhỏ hơn DM2 của y. Trả về 0 nếu DM1 của x lớn hơn DM1 của y hoặc DM2 của x lớn hơn hoặc bằng DM2 của y.

27) Khai báo: `void AddSVMH_1_2(NODE *&T, HocVien X)`

Chức năng: thêm X vào cây nhị phân tìm kiếm T, X là nút trong cây con bên trái T nếu chương trình con: `sosanh` trả về là 1. Và X là nút trong cây con bên phải T nếu chương trình con: `sosanh` trả về là 0.

28) Khai báo: `void Duyet_NLR_3(NODE *x, NODE *&T1)`

Chức năng: Thêm học viên vào cây nhị phân tìm kiếm T1 theo DM1 nếu DM1 bằng nhau thì theo DM2 bằng cách gọi chương trình con: AddSVMH_1_2 và duyệt theo thứ tự trước của cây nhị phân tìm kiếm x.

29) Khai báo: void Tang_dan_DM1_DM2(NODE *T)

Chức năng: Khởi tạo giá trị NULL ban đầu cho cây nhị phân tìm kiếm T1 và gọi các chương trình con: Duyet_NLR_3, Duyet_LNR_3. In ra màn hình tên các cột: Ma so hoc vien, Ho ten hoc vien, Diem mon 1, Diem mon 2.

30) Khai báo: void Duyet_inthongtin(NODE *T, char A[], char &check)

Chức năng: In ra màn hình họ tên học viên và điểm số của các nút trong T có mã môn học bằng chuỗi A. Bằng cách duyệt theo theo thứ tự trước của cây nhị phân tìm kiếm T.

31) Khai báo: void Inthongtin_theoMMH(NODE *T)

Chức năng: dùng để nhập mã môn học cần in thông tin và gọi chương trình con: Duyet_inthongtin.

32) Khai báo: void Viet_NLR(NODE *T)

Chức năng: Dùng để viết ra màn hình mã số học viên, họ tên học viên ở dạng cây lưu trữ.

33) Khai báo: void Ghi_dat(NODE *T, FILE *f)

Chức năng: dùng để ghi dữ liệu các nút của cây nhị phân tìm kiếm T vào file nhị phân f, bằng cách duyệt theo thứ tự trước của cây nhị phân tìm kiếm T. Dữ liệu ghi mỗi nút bao gồm: mã số, họ tên, mã môn học 1, điểm môn học 1, mã môn học 2, điểm môn học 2, mã số và điểm của các môn học phụ, kí tự \n vào cuối.

34) Khai báo: void Ghifile_dat(NODE *T)

Chức năng: dùng để nhập vào đường dẫn, tên file để mở ghi file nhị phân và gọi chương trình con: Ghi_dat.

35) Khai báo: void NewLIST(List *&T, Value A)

Chức năng: Dùng để thêm A vào danh sách T.

36) Khai báo: void Noi_chuoi(char MMH[], char temp)

Chức năng: dùng để thêm kí tự chứa trong temp vào đầu chuỗi MMH.

37) Khai báo: void Mofile_dat()

Chức năng: dùng để nhập vào đường dẫn, tên file nếu chọn 1 hoặc đường dẫn mặc định D:\DATACTDL\hocvien.dat nếu chọn 2, để mở đọc file nhị phân. In ra màn hình menu mở file và cho người dùng nhập lựa chọn và gọi các chương trình con tương ứng.

38) Khai báo: int main()

Chức năng: Dùng để khởi tạo giá trị ban đầu NULL cho cây nhị phân tìm kiếm dx. Và in ra màn hình menu chính, cho người dùng nhập lựa chọn và gọi các chương trình con tương ứng.

HƯỚNG DẪN DÙNG MENU CHÍNH:

Bước 1: Nhập 1 số nguyên từ bàn phím trong phạm vi từ 1-12 và nhấn phím ENTER.

Bước 2:

+ Nhập 1: bạn sẽ nhập thông tin cho một học viên và thêm học viên đó vào cây.

+ Nhập 2: bạn sẽ nhập thông tin môn học phụ cho một học viên theo mã số học viên.

+ Nhập 3: chọn in ra màn hình họ tên học viên, điểm môn 1, điểm môn 2 theo mã số học viên bạn nhập vào.

- + Nhập 4: chọn xoá một học viên trong cây nhị phân tìm kiếm theo mã số học viên bạn nhập vào.
- + Nhập 5: để in ra màn hình mã số học viên, họ tên, điểm môn 1 của các học viên theo thứ tự tăng dần của điểm môn 1.
- + Nhập 6: để in ra màn hình mã số học viên, họ tên, điểm môn 2 của các học viên theo thứ tự tăng dần của điểm môn 2.
- + Nhập 7: để in ra màn hình mã số học viên, họ tên, điểm môn 1, điểm môn 2 được sắp xếp theo thứ tự tăng dần của điểm môn 1, nếu điểm môn 1 bằng nhau thì xếp theo thứ tự tăng dần của điểm môn 2.
- + Nhập 8: để nhập vào một mã số môn học và in ra màn hình họ tên học viên, điểm môn học của những học viên có học môn đó.
- + Nhập 9: để in ra màn hình mã số học viên, tên học viên theo dạng cây lưu trữ, thông tin in ra có dạng: mã số học viên, tên học viên, vị trí trong cây.
- + Nhập 10: để ghi dữ liệu của các nút trong cây nhị phân tìm kiếm vào file nhị phân do bạn nhập được dẫn.
- + Nhập 11: là để đọc dữ liệu từ một file nhị phân mà bạn nhập được dẫn hoặc đường dẫn mặc định D:\\DATACTDL\\hocvien.dat và các dữ liệu đó được lưu vào cây nhị phân tìm kiếm.
- + Nhập 12: để thoát khỏi MENU CHINH và kết thúc chương trình.

HƯỚNG DẪN DÙNG MENU MO FILE:

Bước 1: Nhập 1 số nguyên từ bàn phím trong phạm vi từ 1-9 và nhấn phím ENTER.

Bước 2:

- + Nhập 1: chọn in ra màn hình họ tên học viên, điểm môn 1, điểm môn 2 theo mã số học viên bạn nhập vào.
- + Nhập 2: chọn xoá một học viên trong cây nhị phân tìm kiếm theo mã số học viên bạn nhập vào.
- + Nhập 3: để in ra màn hình mã số học viên, họ tên, điểm môn 1 của các học viên theo thứ tự tăng dần của điểm môn 1.
- + Nhập 4: để in ra màn hình mã số học viên, họ tên, điểm môn 2 của các học viên theo thứ tự tăng dần của điểm môn 2.
- + Nhập 5: để in ra màn hình mã số học viên, họ tên, điểm môn 1, điểm môn 2 được sắp xếp theo thứ tự tăng dần của điểm môn 1, nếu điểm môn 1 bằng nhau thì xếp theo thứ tự tăng dần của điểm môn 2.
- + Nhập 6: để nhập vào một mã số môn học và in ra màn hình họ tên học viên, điểm môn học của những học viên có học môn đó.
- + Nhập 7: để in ra màn hình mã số học viên, tên học viên theo dạng cây lưu trữ, thông tin in ra có dạng: mã số học viên, tên học viên, vị trí trong cây.
- + Nhập 8: để ghi dữ liệu của các nút trong cây nhị phân tìm kiếm vào file nhị phân do bạn nhập được dẫn.
- + Nhập 9: để thoát khỏi MENU MO FILE và trở về MENU CHINH.