**Đề Cương Môn Lập Trình Hướng Đối Tượng C++**

**Đề 1**

**Câu 1 :**

Cài đặt lớp Vector gồm n thành phần là các số thực với các phương thức sau:

+ Hàm tạo mặc định;

+ Hàm hủy;

+ Hàm nhập;

+ Hàm xuất;

+ Hàm toán tử hai Vector;

+ Hàm toán tử \* hai Vector;

+ Hàm toán tử chỉ số.

**b)** Viết hàm main sử dụng lớp Vector đã xây dựng để nhập vào một mảng các Vector có cùng số chiều sau đó:

+ In ra danh sách các Vector vừa nhập;

+ Tính và in ra tổng của các Vector.

Lời giải

#include<iostream>

using namespace std;

//-------------------------------//

class Vector {

int n;

float data[255];

public:

Vector();

Vector(int);

~Vector();

void Nhap();

void Xuat();

Vector operator+(Vector);

Vector operator\*(Vector);

float operator[](int);

};

//-------------------------------//

int main() {

int n;

cout << "Nhap vao so phan tu: ";

cin >> n;

Vector v1(n);

v1.Nhap();

cout << "Vector 1: ";

v1.Xuat();

cout << "\n\*------------------------\*\n";

Vector v2(n);

v2.Nhap();

cout << "Vector 2: ";

v2.Xuat();

cout << "\n\*------------------------\*\n";

Vector v3;

v3 = v1 + v2;

v3.Xuat();

cout << "\n\*------------------------\*\n";

//cout << v3[1];

Vector v4 ;

v4 = v1\*v2;

v4.Xuat();

cout << "\n\*------------------------\*\n";

cout << v4[0];

return 0;

}

//-------------------------------//

Vector::Vector() {

n = 0;

}

//-------------------------------//

Vector::Vector(int num) {

n = num;

}

//-------------------------------//

Vector::~Vector() {

n = -1;

}

//-------------------------------//

void Vector::Nhap() {

for(int i = 0; i < n; i++) {

cout << "V[" << i << "]=";

cin >> data[i];

}

}

//-------------------------------//

void Vector::Xuat() {

for (int i = 0; i < n; i++)

cout << data[i] << "  ";

}

//-------------------------------//

Vector Vector::operator+(Vector param) {

Vector temp(n);

for (int i = 0; i < n; i++)

temp.data[i] = data[i] + param.data[i];

return temp;

}

//-------------------------------//

Vector Vector::operator\*(Vector param) {

Vector temp(n);

for (int i = 0; i < n; i++)

temp.data[i] = data[i] \* param.data[i];

return temp;

}

//-------------------------------//

float Vector::operator[](int index) {

if (index >= n) {

cout << "Khong co phan tu nay";

exit (0);

}

else

return (data[index]);

}

//-------------------------------//

**Câu 2 :**

a) Xây dựng lớp các nhân viên gồm các thuộc tính dữ liệu họ tên, hệ số lương, phụ cấp với các phương thức nhập, xuất và tính lương (lương = hệ số lương \* 1310 + phụ cấp). Kế thừa lớp nhân viên xây dựng lớp Cán bộ có thêm 1 thuộc tính hệ số chức vụ với các phương thức như lớp nhân viên biết lương cán bộ = (hệ số lương + hệ số chức vụ) \* 1310 + phụ cấp.

b) Viết hàm main nhập vào một mảng n Nhân viên và Cán bộ sau đó: + In ra danh sách vừa nhập; + In ra danh sách tối đa 10% nhân viên có thu nhập thấp nhất và có hệ số lương nhỏ hơn 3.0 để nhận trợ cấp khó khăn cuối năm.

Lời giải

#include<iostream>

#include<stdlib.h>

using namespace std;

class Nhanvien

{

protected:

char hoten[30];

float hesoluong;

float phucap;

public:

   void nhap()

   {

    cout<<"Nhap ten : ";

    fflush(stdin);

    cin>>hoten;

    cout<<"Nhap he so luong: ";

    cin>>hesoluong;

    cout<<"Nhap phu cap: ";

    cin>>phucap;

   }

   void xuat()

   {

    cout<<"Ten : "<<hoten<<endl;

    cout<<"He So luong: "<<hesoluong<<endl;

    cout<<"Phu cap: "<<phucap<<endl;

   }

   float luong()

   {

    return(hesoluong\*1310+phucap);

   }

   float hsl()

{

return hesoluong;

}

};

class Canbo:public Nhanvien

{

float hesochucvu;

public:

    void nhap()

    {

     Nhanvien::nhap();

     cout<<"Nhap he so chuc vu: ";

     cin>>hesochucvu;

}

void xuat()

{

Nhanvien::xuat();

cout<<"He so chuc vu: "<<hesochucvu<<endl;

}

float luong()

{

return((hesoluong+hesochucvu)\*1310+phucap);

}

};

int main()

{

int n;

cout<<"Nhap vao so nhan vien: ";

cin>>n;

Nhanvien  nv[100];

Canbo cb[100];

for(int i=0;i<n;i++)

{

cout<<"Nhan vien thu: "<<i+1<<endl;

nv[i].nhap();

    }

    cout<<"---------Danh sach nhan vien vua nhap-------------"<<endl;

    for(int i=0;i<n;i++)

    {

     cout<<"\*-----------------\*"<<endl;

     cout<<"Can Bo thu "<<i+1<<endl;

     nv[i].xuat();

     cout<<"Luong: "<<nv[i].luong()<<endl;

}

/\*for(int i=0;i<n;i++)

{

cout<<"Can Bo thu: "<<i+1<<endl;

cb[i].nhap();

    }

    cout<<"-----------------Danh sach can bo vua nhap---------------"<<endl;

    for(int i=0;i<n;i++)

    {

     cout<<"\*-----------------\*"<<endl;

     cout<<"Nhan vien thu "<<i+1<<endl;

     cb[i].xuat();

     cout<<"Luong: "<<cb[i].luong()<<endl;

}\*/

int dem=1;

for(int i=0;i<n-1;i++)

  for(int j=i+1;j<n;j++)

      if(nv[i].luong()>nv[j].luong())

      {

      Nhanvien tg=nv[i];

      nv[i]=nv[j];

      nv[j]=tg;

  }

    cout<<"Danh sach sap xep luong tang dan :"<<endl;

for(int i=0;i<n;i++)

    {

     cout<<"\*-----------------\*"<<endl;

     cout<<"Nhan vien thu "<<i+1<<endl;

     nv[i].xuat();

     cout<<"Luong: "<<nv[i].luong()<<endl;

}

    cout<<"Nhan vien nhan tro cap la: "<<endl;

float min=nv[0].luong();

for(int i=0;i<n;i++)

{

if (min>nv[i].luong())

  min=nv[i].luong();

}

for(int i=0;i<n;i++)

  if(min==nv[i].luong()&& nv[i].hsl()<3.0)

      {

  nv[i].xuat();

      dem++;

      }

return 0;

}

**Đề 2**

**Câu 1 :**

Xây dựng lớp các hàng hóa gồm các thuộc tính mã hàng hóa (số nguyên), tên hàng hóa, giá nhập, giá bán, số lượng và các phương thức:

+ Hàm tạo mặc định;

+ Hàm hủy;

+ Hàm nhập;

+ Hàm xuất;

+ Hàm tính lợi nhuận = (giá bán – giá mua) \* số lượng bán.

b) Viết hàm main sử dụng lớp hàng hóa đã xây dựng nhập vào một mảng các hàng hóa sau đó:

+ In ra danh sách các hàng hóa vừa nhập;

+ Tính và in ra tổng lợi nhuận của các hàng hóa;

+ In ra danh sách tối đa 10% mặt hàng bán lãi nhất có số lượng bán > 5.

Lời giải

#include<iostream>

using namespace std;

class Hanghoa{

int mahh;

string tenhh;

int gianhap;

int giaban;

int soluong;

public:

    Hanghoa();

~Hanghoa();

void nhap();

void xuat();

int soluongban();

int loinhuan();

};

int main()

{

Hanghoa a[100];

Hanghoa tg;

int n;

cout<<"Nhap so luong hang hoa: ";cin>>n;

for(int i=0;i<n;i++)

{

a[i].nhap();

}

int s=0;

for(int i=0;i<n;i++)

{

a[i].xuat();

cout<<"Loi nhuan la: "<<a[i].loinhuan()<<endl;

s=s+a[i].loinhuan();

}

cout<<"Tong loi nhuan cua cac hang hoa la : "<<s;

for(int i=0;i<n-1;i++)

  for(int j=i+1;j<n;j++)

      if(a[i].loinhuan()<a[j].loinhuan())

      {

      tg=a[i];

      a[i]=a[j];

      a[j]=tg;

  }

cout<<"=======Sap xep giam dan theo loi nhuan========"<<endl;

for(int i=0;i<n;i++)

{

a[i].xuat();

cout<<"Loi nhuan la: "<<a[i].loinhuan()<<endl;

}

    int d=1;

cout<<"Danh sach mat hang co loi nhuan cao nhat"<<endl;

int max=a[0].loinhuan();

for(int i=0;i<n;i++)

  if(max<a[i].loinhuan())

    max=a[i].loinhuan();

for(int i=0;i<n;i++)

  {

  if(max==a[i].loinhuan()&&a[i].soluongban()>5 && d<=n\*0.1)

    {

a[i].xuat();

d++;

}

}

}

Hanghoa::Hanghoa()

{

mahh=0;

tenhh='a';

gianhap=0;

giaban=0;

soluong=0;

}

Hanghoa::~Hanghoa()

{

mahh=-1;

tenhh='z';

gianhap=-1;

giaban=-1;

soluong=-1;

}

void Hanghoa::nhap()

{

cout<<"Nhap ma hang hoa: ";cin>>mahh;

cout<<"Nhap ten hang hoa: ";cin>>tenhh;

cout<<"Nhap gia nhap: ";cin>>gianhap;

cout<<"Nhap gia ban: ";cin>>giaban;

cout<<"Nhap so luong: ";cin>>soluong;

}

void Hanghoa::xuat()

{

    cout<<"ma hang hoa: "<<mahh<<endl;

cout<<"ten hang hoa: "<<tenhh<<endl;

cout<<"gia nhap: "<<gianhap<<endl;

cout<<"gia ban: "<<giaban<<endl;

cout<<"so luong: "<<soluong<<endl;

}

int Hanghoa::loinhuan()

{

return ((giaban-gianhap)\*soluong);

}

int Hanghoa::soluongban()

{

return soluong;

}

**Câu 2 :**

a) Xây dựng lớp các thí sinh xét tuyển đại học gồm các thành viên dữ liệu họ tên, điểm trung bình, điểm tiếng Anh với các phương thức nhập, xuất dữ liệu, xét tuyển (trả về true nếu thí sinh trúng tuyển: điểm trung bình >= 7.0 và điểm tiếng Anh >= 400). Kế thừa từ lớp thí sinh để xây dựng lớp thí sinh ưu tiên có thêm 1 trường dữ liệu là điểm ưu tiên với các phương thức như lớp thí sinh và tiêu chí trúng tuyển là điểm trung bình + điểm ưu tiên >= 7.0 và điểm tiếng Anh >= 400.

b)  Viết hàm main nhập vào một mảng các thí sinh (bình thường và ưu tiên) đã xây dựng ở câu a sau đó:

+ In ra danh sách các thí sinh vừa nhập;

+ In ra danh sách các thí sinh trúng tuyển không vượt quá 30% tổng số thí sinh.

Lời giải

#include<iostream>

using namespace std;

class Thisinh{

protected:

char hoten[30];

float DTB;

float tienganh;

public:

    void nhap()

    {

     cout<<"Nhap ten thi sinh: ";cin>>hoten;

     cout<<"Nhap diem trung binh: ";cin>>DTB;

     cout<<"Nhap diem tieng anh: ";cin>>tienganh;

}

void xuat()

{

cout<<"Ho ten thi sinh: "<<hoten<<endl;

cout<<"Diem trung binh: "<<DTB<<endl;

cout<<"Diem tieng anh: "<<tienganh<<endl;

}

bool xettuyen()

{

return(DTB>=7.0 && tienganh>=400);

}

};

class Uutien:public Thisinh{

float DiemUT;

public:

void nhap()

{

Thisinh::nhap();

cout<<"Nhap diem uu tien: ";cin>>DiemUT;

}

void xuat()

{

Thisinh::xuat();

cout<<"Diem uu tien: "<<DiemUT<<endl;

}

bool xettuyen()

{

return(DTB+DiemUT>=7.0 && tienganh>=400);

}

};

int main()

{

Thisinh a[100];

Uutien b[100];

int n;

cout<<"Nhap tong so thi sinh: ";cin>>n;

for(int i=0;i<n;i++)

{

cout<<"Nhap thong tin cho thi sinh thu "<<i+1<<endl;

a[i].nhap();

}

cout<<"======DANH SACH THI SINH BINH THUONG======"<<endl;

for(int i=0;i<n;i++)

{

cout<<"Thi sinh thu "<<i+1<<endl;

a[i].xuat();

if(a[i].xettuyen())

  cout<<"Trung tuyen"<<endl;

}

cout<<"==========================DANH SACH THI SINH BINH THUONG TRUNG TUYEN======================="<<endl;

int d=1;

for(int i=0;i<n;i++)

{

if(a[i].xettuyen() && d<=n\*0.3)

  {

  a[i].xuat();

  cout<<"DA TRUNG TUYEN"<<endl;

  d++;

  }

}

/\*

cout<<"======DANH SACH THI SINH UU TIEN======"<<endl;

for(int i=0;i<n;i++)

{

cout<<"Thi sinh thu "<<i+1<<endl;

b[i].xuat();

if(b[i].xettuyen()==true)

  cout<<"Trung tuyen"<<endl;

}

cout<<"==========================DANH SACH THI SINH UU TIEN TRUNG TUYEN======================="<<endl;

int dem=1;

for(int i=0;i<n;i++)

{

if(b[i].xettuyen()==true && dem<=n\*0.3)

  {

  b[i].xuat();

  cout<<"DA TRUNG TUYEN"<<endl;

  d++;

  }

}\*/

return 0;

}

**Đề 3**

**Câu 1 :**

Một cửa hàng bán thuốc lưu thông tin các loại thuốc gồm: Tên thuốc, mã thuốc, số lượng và giá bán. Theo quan điểm Lập trình HĐT bạn hãy viết chương trình thực hiện:

a) Nhập và in ra màn hình danh sách các loại thuốc có trong cửa hàng;

b) Sắp xếp danh sách các loại thuốc theo tên, cùng tên theo mã.

Lời giải

#include<iostream>

#include<string.h>

using namespace std;

class Thuoc{

char ten[30];

int mathuoc;

int soluong;

int giaban;

public:

//Thuoc();

//~Thuoc();

void nhap()

{

cout<<"Nhap ma thuoc: ";cin>>mathuoc;

cout<<"Nhap ten thuoc: ";cin>>ten;

cout<<"Nhap so luong: ";cin>>soluong;

cout<<"Nhap gia ban: ";cin>>giaban;

}

void xuat()

{

cout<<"Ma thuoc: "<<mathuoc<<endl;

cout<<"Ten thuoc: "<<ten<<endl;

cout<<"So luong: "<<soluong<<endl;

cout<<"Gia ban: "<<giaban<<endl;

}

void sapxep(Thuoc a[100],int n)

{

Thuoc tg;

for(int i=0;i<n-1;i++)

  for(int j=i+1;j<n;j++)

    if(strcmp(a[i].ten,a[j].ten)>0)

    {

    tg=a[i];

    a[i]=a[j];

    a[j]=tg;

}

else if(strcmp(a[i].ten,a[j].ten)==0 && a[i].mathuoc>a[j].mathuoc)

{

tg=a[i];

    a[i]=a[j];

    a[j]=tg;

}

}

};

int main()

{

Thuoc a[100],tg;

Thuoc b;

int n;

cout<<"Nhap so luong thuoc: ";cin>>n;

for(int i=0;i<n;i++)

{

cout<<"Nhap thong tin cho loai thuoc thu "<<i+1<<endl;

a[i].nhap();

}

cout<<"========Danh sach ban dau========"<<endl;

for(int i=0;i<n;i++)

{

cout<<"Loai thuoc thu "<<i+1<<endl;

a[i].xuat();

}

cout<<"=============Danh sach sau khi sap xep============="<<endl;

b.sapxep(a,n);

for(int i=0;i<n;i++)

{

cout<<"Loai thuoc thu "<<i+1<<endl;

a[i].xuat();

}

return 0;

}

**Câu 2**

**a) Thông tin của các thí sinh thi đại học gồm: tên thí sinh, số báo danh, điểm thi ba môn (môn một, môn hai, môn ba). Hãy cài đặt lớp thí sinh với các phương thức: cấu tử mặc định, cấu tử có tham số, toán tử nhập, xuất và hàm tính tổng điểm (= điểm môn một + điểm môn hai + điểm môn ba).**

**b) Giả sử đối với các thí sinh là con thương binh, liệt sĩ và con em vùng sâu vùng xa thì có thêm điểm ưu tiên và công thức tính tổng điểm là = điểm môn một + điểm môn hai + điểm môn ba + điểm ưu tiên. Xây dựng các lớp trên sử dụng kỹ thuật đa thể và ràng buộc động.**

**Viết chương trình cho phép nhập vào một mảng các thí sinh, in ra tổng điểm của tất cả các thí sinh đó và cho biết thí sinh nào thi đỗ, thi trượt. Điều kiện thi đỗ là không môn nào bị điểm 0 và tổng điểm >= 15.**

Lời giải

#include<iostream>

#include<string.h>

using namespace std;

class Thisinh{

protected:

string ten;

int sbd;

float mon1,mon2,mon3;

public:

    Thisinh()

{

ten='a';

sbd=0;

mon1=0;

mon2=0;

mon3=0;

}

Thisinh(int n)

{

cout<<"";

}

friend  istream &operator>>(istream &input, Thisinh &ts)

{

cout<<"Nhap ten thi sinh: ";input>>ts.ten;

cout<<"Nhap so bao danh: ";input>>ts.sbd;

cout<<"Nhap diem mon1: ";input>>ts.mon1;

cout<<"Nhap diem mon2: ";input>>ts.mon2;

cout<<"Nhap diem mon3: ";input>>ts.mon3;

}

friend  ostream &operator<<(ostream &output, Thisinh &ts)

{

output<<"Ten: "<<ts.ten<<endl;

output<<"So bao danh: "<<ts.sbd<<endl;

output<<"Diem mon 1: "<<ts.mon1<<endl;

output<<"Diem mon 2: "<<ts.mon2<<endl;

output<<"Diem mon 3: "<<ts.mon3<<endl;

}

virtual int tongdiem()

{

return (mon1+mon2+mon3);

}

};

class ThisinhUT:public Thisinh{

float DiemUT;

public:

friend istream &operator>>(istream &input, ThisinhUT &ts)

{

cout<<"Nhap ten thi sinh: ";input>>ts.ten;

cout<<"Nhap so bao danh: ";input>>ts.sbd;

cout<<"Nhap diem mon1: ";input>>ts.mon1;

cout<<"Nhap diem mon2: ";input>>ts.mon2;

cout<<"Nhap diem mon3: ";input>>ts.mon3;

cout<<"Nhap diem Uu tien: ";input>>ts.DiemUT;

}

friend ostream &operator<<(ostream &output, ThisinhUT &ts)

{

output<<"Ten: "<<ts.ten<<endl;

output<<"So bao danh: "<<ts.sbd<<endl;

output<<"Diem mon 1: "<<ts.mon1<<endl;

output<<"Diem mon 2: "<<ts.mon2<<endl;

output<<"Diem mon 3: "<<ts.mon3<<endl;

output<<"Diem Uu tien: "<<ts.DiemUT<<endl;

}

int tongdiem()

{

return(mon1+mon2+mon3+DiemUT);

}

    bool thido()

    {

     return (mon1+mon2+mon3+DiemUT>=15 && mon1!=0 && mon2!=0 && mon3!=0);

}

};

int main()

{

Thisinh \*ts=new ThisinhUT;

ts->tongdiem();

ThisinhUT a[100];

int n;

cout<<"Nhap vao so thi sinh: "<<endl;cin>>n;

for (int i=0;i<n;i++)

{

cout<<"Nhap thong tin cho thi sinh thu "<<i+1<<endl;

cin>>a[i];

}

for (int i=0;i<n;i++)

{

cout<<"Thong tin cho thi sinh thu "<<i+1<<endl;

cout<<a[i];

cout<<"Tong diem: "<<a[i].tongdiem()<<endl;

if(a[i].thido())

  cout<<"Thi do"<<endl;

else

  cout<<"Truot"<<endl;

}

return 0;

}

**Đề 4**

**Câu 1 :**

Thông tin của nhân viên trong một công ty gồm: Họ tên, Năm sinh, Lương cơ bản (lương một ngày công), Số ngày công.

a) Anh (chị) hãy xây dựng lớp Nhan\_Vien của công ty với các thuộc tính trên. Lớp Nhan\_Vien có các phương thức: Khởi tạo, Nhập thông tin, In thông tin, Tính lương tháng của nhân viên (theo công thức: lương tháng = số ngày công \* lương cơ bản).

b) Xây dựng lớp Bang\_Luong với các thuộc tính: Số lượng nhân viên, Danh sách nhân viên. Lớp Bang\_Luong có các phương thức: Khởi tạo (Khởi tạo một danh sách rỗng), Hủy bỏ (hủy danh sách), Nhập danh sách, In danh sách, Thống kế các nhân viên có lương tháng < 5.000.000.

Lời giải

#include<iostream>

#include<string.h>

using namespace std;

class Nhanvien{

string hoten;

int namsinh;

int luongcb;

int songaycong;

public:

    Nhanvien()

{

hoten='a';

namsinh=0;

luongcb=0;

songaycong=0;

}

void nhap()

{

cout<<"Nhap ten nhan vien: ";cin>>hoten;

cout<<"Nhap nam sinh: ";cin>>namsinh;

cout<<"Nhap luong co ban: ";cin>>luongcb;

cout<<"Nhap so ngay cong: ";cin>>songaycong;

}

void xuat()

{

cout<<"Ho ten: "<<hoten<<endl;

cout<<"Nam sinh: "<<namsinh<<endl;

cout<<"luong co ban: "<<luongcb<<endl;

cout<<"So ngay cong: "<<songaycong<<endl;

}

float luong1thang()

{

return (luongcb\*songaycong);

}

};

class Bangluong{

int soNV;

Nhanvien \*nv[100];

public:

    Bangluong()

{

nv[100]=NULL;

}

~Bangluong()

{

delete nv[100];

}

void nhap()

{

cout<<"Nhap so luong nhan vien: ";cin>>soNV;

for(int i=0;i<soNV;i++)

  {

  cout<<"Nhap thong tin cua nhan vien thu "<<i+1<<endl;

  nv[i]=new Nhanvien;

  nv[i]->nhap();

      }

}

void xuat()

{

for(int i=0;i<soNV;i++)

  {

  cout<<"Nhan vien thu "<<i+1<<endl;

    nv[i]->xuat();

}

}

void thongke()

{

for(int i=0;i<soNV;i++)

  if(nv[i]->luong1thang()<5000000)

      {

      //nv[i]=new Nhanvien;

  nv[i]->xuat();

    }

}

};

int main()

{

Bangluong a;

a.nhap();

cout<<"DANH SACH BAN DAU "<<endl;

a.xuat();

cout<<"DANH SACH NHAN VIEN CO LUONG <5.000.000"<<endl;

a.thongke();

return 0;

}

**Câu 2 :**

**a) Xây dựng lớp cơ sở trừu tượng mặt hàng (mô tả các mặt hàng nói chung) gồm các thuộc tính dữ liệu: tên mặt hàng, mã mặt hàng, giá nhập, giá xuất và phương thức: tính thuế, khởi tạo, nhập, xuất.**

**b) Giả sử đối với mặt hàng gia dụng nhập khẩu, ô tô nhập khẩu và máy móc nhập khẩu thì thuế nhập khẩu lần lượt là 10%, 30%, 12% của giá nhập. Hãy cài đặt các lớp trên sử dụng kỹ thuật đa thể và ràng buộc động. Viết chương trình cho phép nhập và in ra danh sách n mặt hàng. Cho biết mặt hàng nào có lãi cao nhất.**

Lời giải

#include<iostream>

#include<string.h>

using namespace std;

class Mathang{ //class chua ham ao thuc su la lop co so truu tuong

protected:

string ten;

int maMH;

float gianhap;

float giaxuat;

public:

    Mathang()

{

ten='a';

maMH=0;

gianhap=0;

giaxuat=0;

}

void nhap()

{

cout<<"Nhap ten mat hang: ";cin>>ten;

cout<<"Nhap ma mat hang: ";cin>>maMH;

cout<<"Gia nhap: ";cin>>gianhap;

cout<<"Gia xuat: ";cin>>giaxuat;

}

void xuat()

{

cout<<"TEN mat hang: "<<ten<<endl;

cout<<"Ma mat hang: "<<maMH<<endl;

cout<<"Gia nhap: "<<gianhap<<endl;

cout<<"Gia xuat: "<<giaxuat<<endl;

}

virtual float tinhthue()=0; //ham ao thuc su

virtual lai()

{

return(giaxuat-gianhap);

}

};

class Giadung: public Mathang{

public:

float tinhthue()

{

return((giaxuat-gianhap)\*(10/100));

}

};

class Oto:public Mathang{

public:

float tinhthue()

{

return((giaxuat-gianhap)\*(30/100));

}

};

class Maymoc:public Mathang{

public:

float tinhthue()

{

return((giaxuat-gianhap)\*(12/100));

}

};

int main()

{

Mathang \*mh[100]; //khong khai bao duoc nhu nay vi trong class Mathang co chua 1 ham ao thuc su

Mathang \*a[100];

Mathang \*b[100];

Mathang \*c[100];

int n;

cout<<"Nhap so luong mat hang: ";cin>>n;

for(int i=0;i<n;i++)

{

a[i]=new Giadung;

cout<<"Nhap thong tin cho mat hang thu "<<i+1<<endl;

a[i]->nhap();

}

for(int i=0;i<n;i++)

{

cout<<"Thong tin mat hang thu "<<i+1<<endl;

a[i]->xuat();

}

float max=0;

for(int i=0;i<n;i++)

{

if(max<(a[i]->lai()-a[i]->tinhthue()))

  max=(a[i]->lai()-a[i]->tinhthue());

}

cout<<"Mat hang co so lai cao nhat la: "<<endl;

for(int i=0;i<n;i++)

if(max==(a[i]->lai()-a[i]->tinhthue()))

    a[i]->xuat();

}

**Đề 5**

**Câu 1 :**

Cho thông tin của một sinh viên gồm: Tên sinh viên, mã sinh viên, điểm trung bình. Bạn hãy xây dựng các lớp cần thiết để có thể:

 a) Nhập, in danh sách gồm n sinh viên.

b) Tìm sinh viên có điểm trung bình lớn nhất. Sắp xếp sinh viên theo tên, cùng tên theo mã sinh viên.

Lời giải

#include<iostream>

#include<string.h>

using namespace std;

class Sinhvien{

char ten[30];

int masv;

float DTB;

public:

    void nhap()

{

cout<<"Ten sinh vien: ";cin>>ten;

cout<<"Ma sinh vien: ";cin>>masv;

cout<<"Diem trung binh: ";cin>>DTB;

}

void xuat()

{

cout<<"Ten sinh vien: "<<ten<<endl;

cout<<"Ma sinh vien: "<<masv<<endl;

cout<<"Diem trung binh: "<<DTB<<endl;

}

void maxdiem(Sinhvien a[],int n)

{

float max=0;

int i;

for( i=0;i<n;i++)

  if(max<a[i].DTB)

    max=a[i].DTB;

for(i=0;i<n;i++)

{

if(max==a[i].DTB)

    {

    a[i].xuat();

    }

    }

}

void sapxep(Sinhvien a[],int n)

{

int i,j;

Sinhvien tg;

for(i=0;i<n-1;i++)

  for(j=i+1;j<n;j++)

      if(strcmp(a[i].ten,a[j].ten)>0)

      {

      tg=a[i];

      a[i]=a[j];

      a[j]=tg;

  }

  else if(strcmp(a[i].ten,a[j].ten)==0 && a[i].masv>a[j].masv)

  {

  tg=a[i];

      a[i]=a[j];

      a[j]=tg;

  }

}

};

int main()

{

Sinhvien a[100],b;

int n;

cout<<"Nhap tong so sinh vien: ";cin>>n;

for(int i=0;i<n;i++)

{

cout<<"Nhap thong tin cho sinh vien thu "<<i+1<<endl;

a[i].nhap();

}

for(int i=0;i<n;i++)

{

cout<<"Thong tin sinh vien thu "<<i+1<<endl;

a[i].xuat();

}

cout<<"Thi sinh co diem trung binh lon nhat: "<<endl;

b.maxdiem(a,n);

cout<<"Danh sach sap xep"<<endl;

b.sapxep(a,n);

for(int i=0;i<n;i++)

{

cout<<"Thong tin sinh vien thu "<<i+1<<endl;

a[i].xuat();

}

}

**Câu 2 :**

a) Một bệnh viện quản lý thông tin bệnh nhân nội trú (họ tên, mã hồ sơ, năm sinh, chuẩn đoán bệnh, tên khoa điều trị, mã phòng+giường) và ngoại trú (họ tên, mã hồ sơ, năm sinh, chuẩn đoán bệnh, tên khoa điều trị, đơn thuốc). Xây dựng các lớp phù hợp có sử dụng kỹ thuật thừa kế.

b) Cài đặt các lớp khác (nếu cần) để có thể xây dựng chương trình nhập và in ra danh sách n bệnh nhân gồm cả nội trú và ngoại trú. Sắp xếp danh sách đó theo mã hồ sơ.

Lời giải

#include<iostream>

#include<stdlib.h>

using namespace std;

class Benhnhan

{

protected:

char hoten[30];

int mahs;

char namsinh[30];

char chuandoanbenh[30];

char tenkhoadieutri[30];

public:

   virtual void nhap()

{

  cout<<"Nhap ma ho so: ";

  cin>>mahs;

  cout<<"Nhap Ho ten benh nhan: ";

  fflush(stdin);

  cin>>hoten;

  cout<<"Nhap nam sinh: ";

  fflush(stdin);

  cin>>namsinh;

  cout<<"Nhap chuan doan benh : ";

  fflush(stdin);

  cin>>chuandoanbenh;

  cout<<"Nhap ten khoa khoa dieu tri: ";

  fflush(stdin);

  cin>>tenkhoadieutri;

}

virtual void xuat()

{

cout<<"Ma ho so: "<<mahs<<endl;

cout<<"Ho ten benh nhan: "<<hoten<<endl;

cout<<"Nam sinh benh nhan: "<<namsinh<<endl;

cout<<"Chuan doan: "<<chuandoanbenh<<endl;

cout<<"Ten khoa dieu tri: "<<tenkhoadieutri<<endl;

}

virtual int set()

{

return mahs;

}

};

class Noitru:public Benhnhan

{

   int maphong;

   int giuong;

public:

   void nhap()

   {

    Benhnhan::nhap();

    cout<<"Nhap ma phong: ";

    cin>>maphong;

    cout<<"Nhap so giuong: ";

    cin>>giuong;

   }

   void xuat()

   {

    Benhnhan::xuat();

    cout<<"Ma phong: "<<maphong<<endl;

    cout<<"Giuong so: "<<giuong<<endl;

}

};

class Ngoaitru:public Benhnhan

{

char donthuoc[30];

public:

void nhap()

{

Benhnhan::nhap();

cout<<"Nhap don thuoc: ";

fflush(stdin);

cin>>donthuoc;

}

void xuat()

{

Benhnhan::xuat();

cout<<"Don thuoc gom: "<<donthuoc<<endl;

}

};

int main()

{

int n;

cout<<"Nhap tong so benh nhan: ";

cin>>n;

Benhnhan \*bn[100];

Benhnhan \*tg;

int chon;

for(int i=0;i<n;i++)

{

cout<<"Moi ban nhap lua chon (1-Noi tru, 2-Ngoai tru)";cin>>chon;

if(chon==1)

  bn[i]=new Noitru;

else if(chon==2)

  bn[i]=new Ngoaitru;

cout<<"Nhap thong tin cho benh nhan thu: "<<i+1<<endl;

bn[i]->nhap();

}

for(int i=0;i<n-1;i++)

  for(int j=i+1;j<n;j++)

      if(bn[i]->set()>bn[j]->set())

      {

      tg=bn[i];

      bn[i]=bn[j];

      bn[j]=tg;

  }

    cout<<"---------------------------------Danh sach Benh Nhan------------------------------";

for(int i=0;i<n;i++)

{

cout<<"\*-----------------\*"<<endl;

bn[i]->xuat();

}

return 0;

}

**Đề 6**

**Câu 1 :**

Xây dựng lớp các ma trận các số nguyên có kích thước 2x2 với các phương thức:

+ Hàm nhập;

+ Hàm xuất;

+ Hàm tính định thức;

+ Hàm cộng hai ma trận;

+ Hàm toán tử chỉ số.

Viết hàm main nhập vào ma trận sau đó:

+ In ra các ma trận vừa nhập;

+ In ra tổng của các ma trận.

Lời giải

#include<iostream>

#include<string.h>

using namespace std;

class Matran{

int a[2][2];

public:

    void nhap()

    {

    for(int i=0;i<2;i++)

for(int j=0;j<2;j++)

  {

  cout<<"Nhap a"<<"["<<i<<"]"<<"["<<j<<"]: ";

  cin>>a[i][j];

}

}

void xuat()

{

for(int i = 0; i < 2; i++)

        {

            for (int j = 0; j < 2; j++)

            {

                cout << a[i][j] << " ";

            }

            cout << endl;

        }

}

int dinhthuc()

{

return(a[0][0]\*a[1][1]-a[0][1]\*a[1][0]);

}

Matran operator+(Matran b)

{

Matran c;

for(int i=0;i<2;i++)

  for(int j=0;j<2;j++)

    c.a[i][j]=a[i][j]+b.a[i][j];

return c;

}

int &operator[](int index)

{

if(index>1 || index<0)

  cout<<"Chi so vua nhap bi sai"<<endl;

else

{

      return a[index][index];

    }

}

};

int main()

{

Matran a[100];

int n;

cout<<"Nhap vao so ma tran ";cin>>n;

for(int i=0;i<n;i++)

{

cout<<"Ma tran thu "<<i+1<<endl;

a[i].nhap();

}

Matran s=a[0];

for(int i=0;i<n;i++)

{

cout<<"Ma tran thu "<<i+1<<endl;

a[i].xuat();

    }

    for(int i=1;i<n;i++)

    {

     s=s+a[i];

}

cout<<"Tong cac ma tran vua nhap la"<<endl;

s.xuat();

//tinh dinh thuc

cout<<a[0].dinhthuc();

//toan tu chi so

//cout<<a[1][0];

return 0;

}

**Câu 2**

a) Cài đặt lớp thẻ thư viện gồm các thông tin mã thẻ, số ngày mượn, phí mượn theo ngày và các phương thức khởi tạo, nhập dữ liệu, xuất dữ liệu, tính phí mượn = số ngày mượn \* phí mượn. Kế thừa lớp thẻ thư viện để xây dựng lớp thẻ ưu tiên có thêm tỉ lệ giảm giá và các phương thức như lớp thẻ thư viện và tính phí mượn  = số ngày mượn \* phí mượn \* (1 - tỉ lệ giảm giá).

 b) Viết hàm main nhập vào một mảng các thẻ thư viện (loại thường và loại ưu tiên) sau đó:

+ In ra danh sách thẻ vừa nhập;

+ In ra danh sách tối đa 10% số thẻ có phí mượn cao nhất và số ngày mượn > 10.

Lời giải

#include<iostream>

using namespace std;

class Thethuvien{

protected:

int mathe;

int songaymuon;

float phimuon;

public:

    Thethuvien()

{

mathe=0;

songaymuon=0;

phimuon=0;

}

virtual void nhap()

{

cout<<"Nhap ma the: ";cin>>mathe;

cout<<"Nhap so ngay muon: ";cin>>songaymuon;

cout<<"Nhap phi muon : ";cin>>phimuon;

}

virtual void xuat()

{

cout<<"Ma the: "<<mathe<<endl;

cout<<"So ngay muon: "<<songaymuon<<endl;

cout<<"Phi muon: "<<phimuon<<endl;

}

virtual float tinhphi()

{

return songaymuon\*phimuon;

}

virtual int ngaymuon()

{

return songaymuon;

}

};

class Theuutien:public Thethuvien{

float giamgia;

public:

    void nhap()

    {

    Thethuvien::nhap();

    cout<<"Nhap ty le giam gia: ";cin>>giamgia;

  }

void xuat()

{

Thethuvien::xuat();

cout<<"Ty le giam gia: "<<giamgia<<endl;

  }

float tinhphi()

{

return (songaymuon\*phimuon\*(1-giamgia));

}

int ngaymuon()

{

return songaymuon;

}

};

int main()

{

int n;

cout<<"Nhap so luong the: ";cin>>n;

Thethuvien \*a[100];

int chon;

for(int i=0;i<n;i++)

{

cout<<"ban muon nhap loai the(1-thuong,2-uu tien): ";cin>>chon;

if(chon==1)

  a[i]=new Thethuvien;

else if(chon==2)

  a[i] =new Theuutien;

else

  break;

cout<<"Nhap thong tin cho loai the thu "<<i+1<<endl;

a[i]->nhap();

}

for(int i=0;i<n;i++)

{

cout<<"Thong tin cho the thu "<<i+1<<endl;

a[i]->xuat();

}

cout<<"=============DANH SACH THE CO PHI MUON CAO NHAT================"<<endl;

float max=a[0]->tinhphi();

int d=1;

for(int i=0;i<n;i++)

  if(max<a[i]->tinhphi() && a[i]->ngaymuon()>10)

    {

max=a[i]->tinhphi();

    }

    for(int i=0;i<n;i++)

      if(max==a[i]->tinhphi()&& a[i]->ngaymuon()>10 &&d<=n\*0.1)

         {

a[i]->xuat();

d++;

    }

return 0;

}

**Đề 7**

**Câu 1**

Xây dựng lớp các hình tròn gồm các thuộc tính dữ liệu là tọa độ tâm và bán kính với các phương thức:

+ Hàm khởi tạo mặc định;

 + Hàm nhập:

+ Hàm xuất;

+ Hàm tính diện tích;

+ Hàm kiểm tra một điểm có nằm trong đường tròn hay không

Nhập vào một mảng các hình tròn và 1 điểm sau đó:

+ In ra mảng các hình tròn vừa nhập;

+ Sắp xếp các hình tròn theo chiều giảm dần của diện tích; + In ra tối đa 10% các hình tròn có diện tích lớn nhất và không chứa điểm đã nhập.

Lời giải

#include<iostream>

#include<math.h>

using namespace std;

class Hinhtron{

int x,y;

float r;

public:

   Hinhtron()

   {

    x=y=0;

    r=0;

}

void nhap()

{

  cout<<"Nhap x: ";cin>>x;

  cout<<"Nhap y: ";cin>>y;

  cout<<"Nhap ban kinh: ";cin>>r;

}

void xuat()

{

cout<<"Toa do tam I("<<x<<","<<y<<")"<<endl;

cout<<"Ban kinh r= "<<r<<endl;

}

float dientich()

{

return 3.14\*r\*r;

}

int check(int x1,int y1)//x1 va y1 la diem kiem tra

{

float a=sqrt((x1-x)\*(x1-x)+(y1-y)\*(y1-y));

if(a<=r)

return 1;

else

return 0;

}

};

int main()

{

int n,x,y;

cout<<"Nhap so hinh tron ";cin>>n;

Hinhtron a[100];

cout<<"Nhap toa do diem can kiem tra"<<endl;

cout<<"Nhap x: ";cin>>x;

cout<<"Nhap y: ";cin>>y;

for(int i=0;i<n;i++)

{

cout<<"Nhap thong tin cho hinh tron thu "<<i+1<<endl;

a[i].nhap();

}

for(int i=0;i<n;i++)

{

cout<<"HInh tron thu "<<i+1<<endl;

a[i].xuat();

a[i].check(x,y);

}

cout<<"DANH SACH SAP XEP THEO CHIEU GIAM DAN CUA DIEN TICH"<<endl;

for(int i=0;i<n-1;i++)

  for(int j=i+1;j<n;j++)

      if(a[i].dientich()<a[j].dientich())

      {

      Hinhtron tg=a[i];

              a[i]=a[j];

              a[j]=tg;

  }

float max=a[0].dientich();

int d=1;

for(int i=0;i<n;i++)

  if(max<a[i].dientich()&& a[i].check(x,y)==0)

      max=a[i].dientich();

cout<<"=======Danh sach hinh tron co dien tich lon nhat ko vuot qua 10% tong so==========="<<endl;

for(int i=0;i<n;i++)

  if(max==a[i].dientich()&& a[i].check(x,y)==0 && d<=n\*0.1)

      {

  a[i].xuat();

  d++;

      }

return 0;

}

**Câu 2**

a) Xây dựng lớp công nhân gồm các thuộc tính dữ liệu họ tên, hệ số lương, phụ cấp và các phương thức nhập, xuất, tính lương = hệ số lương \* 1310 + phụ cấp, tính thưởng  = 3 \* lương. Kế thừ từ lớp công nhân xây dựng lớp cán bộ với các phương thức như lớp công nhân nhưng có thêm 1 thuộc tính là phụ cấp đặc thù và lương = hệ số lương \* 1310 + phụ cấp + phụ cấp đặc  thù, thưởng = 3.5 \* lương.

b) Viết hàm main nhập vào một mảng n công nhân và cán bộ sau đó: + In ra mảng vừa nhập; + In ra danh sách tối đa 20% công nhân hoặc cán bộ có lương thấp nhất và hệ số lương  < 3 để nhận hỗ trợ.

Lời giải

#include<iostream>

using namespace std;

class Congnhan{

protected:

char hoten[30];

float hsl;

float phucap;

public:

   virtual void nhap()

   {

    cout<<"Nhap ten nhan vien: ";cin>>hoten;

    cout<<"Nhap he so luong: ";cin>>hsl;

    cout<<"Nhap phu cap: ";cin>>phucap;

}

virtual void xuat()

{

cout<<"Ten nhan vien: "<<hoten<<endl;

cout<<"He so luong: "<<hsl<<endl;

cout<<"Phu cap: "<<phucap<<endl;

}

virtual float luong()

{

return hsl\*1310+phucap;

}

virtual float thuong()

{

return 3\*luong();

}

virtual float hesoluong()

{

return hsl;

}

};

class Canbo:public Congnhan{

float phucapdacthu;

public:

void nhap()

{

Congnhan::nhap();

cout<<"Nhap phu cap dac thu: ";cin>>phucapdacthu;

}

void xuat()

{

Congnhan::xuat();

cout<<"Phu cap dac thu: "<<phucapdacthu<<endl;

}

float luong()

{

return hsl\*1310+phucap+phucapdacthu;

}

float thuong()

{

return 3.5\*luong();

}

float hesoluong()

{

return hsl;

}

};

int main()

{

int n;

cout<<"Nhap tong so cong nhan , can bo: ";cin>>n;

Congnhan \*a[100];

int chon;

for(int i=0;i<n;i++)

{

cout<<"Nhap (1-cong nhan ,2-Can bo): ";cin>>chon;

if(chon==1)

  a[i]=new Congnhan;

else if(chon==2)

  a[i]=new Canbo;

else

  break;

cout<<"Nhap thong tin cho Nhan su thu "<<i+1<<endl;

a[i]->nhap();

}

for(int i=0;i<n;i++)

{

cout<<"Thong tin Nhan su thu "<<i+1<<endl;

a[i]->xuat();

}

cout<<"==============DANH SACH CONG NHAN HOAC CAN BO CO LUONG THAP NHAT TOI DA 20% TONG SO==============="<<endl;

float min=a[0]->luong();

for(int i=0;i<n;i++)

  if(min>a[i]->luong()&&a[i]->hesoluong()<3)

    min=a[i]->luong();

int d=1;

for(int i=0;i<n;i++)

  if(min==a[i]->luong()&&a[i]->hesoluong()<3&&d<n\*0.2)

      {

      a[i]->xuat();

      d++;

  }

return 0;

}

**Đề 8**

**Câu 1 :**

Cài đặt lớp học viên gồm các thuộc tính dữ liệu mã học viên, họ tên, ngành học và các phương thức:

+ Hàm tạo mặc định và hàm tạo có tham số;

 + Hàm nhập;

+ Hàm xuất.

Hãy nhập vào một mảng n học viên và sau đó:

+ In ra danh sách học viên vừa nhập;

+ Nhập vào một ngành học và in ra danh sách các học viên học theo ngành đó;

+ Sắp xếp danh sách học viên tăng dần theo họ tên, cùng họ tên theo ngành học và in kết quả ra màn hình.

Lời giải

#include<iostream>

#include<string.h>

using namespace std;

class Hocvien{

int mahv;

string ten;

string nganhhoc;

public:

    Hocvien()

{

ten="a";

nganhhoc="b";

mahv=0;

}

Hocvien(int x,string a,string b)

{

mahv=x;

ten=a;

nganhhoc=b;

}

void nhap()

{

cout<<"Nhap ma hoc vien: ";cin>>mahv;

cout<<"Nhap ten hoc vien: ";cin>>ten;

cout<<"Nhap nganh hoc: ";cin>>nganhhoc;

}

void xuat()

{

cout<<"Ma hoc vien: "<<mahv<<endl;

cout<<"Ten hoc vien: "<<ten<<endl;

cout<<"Nganh hoc: "<<nganhhoc<<endl;

}

string NH()

{

return nganhhoc;

}

string hoten()

{

return ten;

}

};

int main()

{

    //Hocvien a(5,"thai","cntt");

    //a.nhap();

//a.xuat();

int n;

cout<<"Nhap vao so hoc vien: ";cin>>n;

Hocvien a[100];

for(int i=0;i<n;i++)

{

cout<<"Nhap thong tin cho hoc vien thu "<<i+1<<endl;

a[i].nhap();

}

cout<<"=====================================DANH SACH BAN DAU==========================================="<<endl;

for(int i=0;i<n;i++)

{

cout<<"Hoc vien thu "<<i+1<<endl;

a[i].xuat();

}

string c;

int d=0;

cout<<"Nhap nganh hoc: ";cin>>c;

cout<<"============DANH SACH HOC VIEN THEO HOC NGANH================"<<c<<endl;

for(int i=0;i<n;i++)

{

if(a[i].NH()==c)

{

  a[i].xuat();

  d++;

    }

}

Hocvien tg;

if(d==0) cout<<"khong co sinh vien nao theo hoc nganh do"<<endl;

cout<<"=====DANH SACH SAP XEP TANG DAN THEO TEN,CUNG TEN THEO NGANH HOC====="<<endl;

for(int i=0;i<n-1;i++)

  for(int j=i+1;j<n;j++)

      if(a[i].hoten()>a[j].hoten())

      {

      tg=a[i];

      a[i]=a[j];

      a[j]=tg;

  }

  else if(a[i].hoten()==a[j].hoten() && a[i].NH()>a[j].NH())

  {

  tg=a[i];

      a[i]=a[j];

      a[j]=tg;

  }

for(int i=0;i<n;i++)

{

cout<<"Hoc vien thu "<<i+1<<endl;

a[i].xuat();

}

return 0;

}

**Câu 2**

a) Xây dựng lớp các thí sinh xét tuyển đại học gồm các thành viên dữ liệu họ tên, điểm trung bình, điểm tiếng Anh với các phương thức nhập, xuất dữ liệu, xét tuyển (trả về true nếu thí sinh trúng tuyển: điểm trung bình >= 7.0 và điểm tiếng Anh >= 400). Kế thừa từ lớp thí sinh để xây dựng lớp thí sinh ưu tiên có thêm 1 trường dữ liệu là điểm ưu tiên với các phương thức như lớp thí sinh và tiêu chí trúng tuyển là điểm trung bình + điểm ưu tiên >= 7.0 và điểm tiếng Anh >= 400.

b) Viết hàm main nhập vào một mảng các thí sinh (bình thường và ưu tiên) đã xây dựng ở câu a sau đó:

+ In ra danh sách các thí sinh vừa nhập;

+ In ra danh sách các thí sinh trúng tuyển không vượt quá 30% tổng số thí sinh.

Lời giải

#include<iostream>

#include<string.h>

using namespace std;

class Hocvien{

int mahv;

string ten;

string nganhhoc;

public:

    Hocvien()

{

ten="a";

nganhhoc="b";

mahv=0;

}

Hocvien(int x,string a,string b)

{

mahv=x;

ten=a;

nganhhoc=b;

}

void nhap()

{

cout<<"Nhap ma hoc vien: ";cin>>mahv;

cout<<"Nhap ten hoc vien: ";cin>>ten;

cout<<"Nhap nganh hoc: ";cin>>nganhhoc;

}

void xuat()

{

cout<<"Ma hoc vien: "<<mahv<<endl;

cout<<"Ten hoc vien: "<<ten<<endl;

cout<<"Nganh hoc: "<<nganhhoc<<endl;

}

string NH()

{

return nganhhoc;

}

string hoten()

{

return ten;

}

};

int main()

{

    //Hocvien a(5,"thai","cntt");

    //a.nhap();

//a.xuat();

int n;

cout<<"Nhap vao so hoc vien: ";cin>>n;

Hocvien a[100];

for(int i=0;i<n;i++)

{

cout<<"Nhap thong tin cho hoc vien thu "<<i+1<<endl;

a[i].nhap();

}

cout<<"=====================================DANH SACH BAN DAU==========================================="<<endl;

for(int i=0;i<n;i++)

{

cout<<"Hoc vien thu "<<i+1<<endl;

a[i].xuat();

}

string c;

int d=0;

cout<<"Nhap nganh hoc: ";cin>>c;

cout<<"============DANH SACH HOC VIEN THEO HOC NGANH================"<<c<<endl;

for(int i=0;i<n;i++)

{

if(a[i].NH()==c)

{

  a[i].xuat();

  d++;

    }

}

Hocvien tg;

if(d==0) cout<<"khong co sinh vien nao theo hoc nganh do"<<endl;

cout<<"=====DANH SACH SAP XEP TANG DAN THEO TEN,CUNG TEN THEO NGANH HOC====="<<endl;

for(int i=0;i<n-1;i++)

  for(int j=i+1;j<n;j++)

      if(a[i].hoten()>a[j].hoten())

      {

      tg=a[i];

      a[i]=a[j];

      a[j]=tg;

  }

  else if(a[i].hoten()==a[j].hoten() && a[i].NH()>a[j].NH())

  {

  tg=a[i];

      a[i]=a[j];

      a[j]=tg;

  }

for(int i=0;i<n;i++)

{

cout<<"Hoc vien thu "<<i+1<<endl;

a[i].xuat();

}

return 0;

}