TRƯỜNG ĐẠI HỌC GIAO THÔNG VÂN TẢI

KHOA CÔNG NGHỆ THÔNG TIN

---------------o0o---------------



**BÀI TẬP LỚN MÔN HỌC**

**CẤU TRÚC DỮ LIỆU VÀ GIẢI THUẬT**

**CHỦ ĐỀ SỐ 26**

|  |  |  |
| --- | --- | --- |
| Giảng viên hướng dẫn | : | Phạm Đình Phong |
| Sinh viên thực hiện | : | Nhữ Đình Đức |
| Lớp  Khóa | :  : | CNTT6  K62 |
| Mã sinh viên | : | 211202522 |

*Hà Nội, tháng 11 năm 2022.*

*Hà Nội, tháng 11 năm 2022*

**Hà Nội, tháng 11 năm 2022**

**MỤC LỤC**

[**A . Đề bài 3**](#_Toc120015679)

[**B. Bài 1 3**](#_Toc120015680)

[***1. Sourses code* 3**](#_Toc120015681)

[***2.Ý tưởng và mô tả thuật toán( thuộc tính ,phương thức các lớp và thời gian chạy phương thức)* 7**](#_Toc120015682)

[**C. Bài 2 10**](#_Toc120015683)

[***1.Full sourses* 10**](#_Toc120015684)

[***2.Ý tưởng và mô tả thuật toán( thuộc tính ,phương thức các lớp và thời gian chạy phương thức)* 12**](#_Toc120015685)

[**D. Bài 3 12**](#_Toc120015686)

[***2.Ý tưởng và mô tả thuật toán( thuộc tính ,phương thức các lớp )* 18**](#_Toc120015687)

[**E. Tài liệu tham khảo 20**](#_Toc120015688)

# **A . Đề bài**

**Bài 26**

1. Cài đặt cấu trúc dữ liệu trừu tượng vector có bộ lặp xuôi và ngược

2. Cài đặt class câu hỏi trắc nghiệm mỗi câu hỏi gồm câu hỏi dạng văn bản và 4 phương án trong đó có 1 phương án đúng

3. Cài đặt class thi dùng cấu trúc vector đã xây dựng để đọc đữ liệu từ câu hỏi từ file văn bản lưu vào vector. Cho phép tạo ra đề thi gồm k câu hỏi ngẫu nhiên trong vector các câu hỏi, khi hiển thị sáo trộn ví trí các phương án trả lời cho phép người thi và tính điểm.

# **B. Bài 1**

## ***1. Sourses code***

//#ifndef VECTOR\_H\_

#define VECTOR\_H\_

#include<algorithm>

#include<cassert>

#include<iostream>

template <class T>

class reverse\_iterator

{

T \*curr;

public:

reverse\_iterator(T \*c=0) {curr=c;}

reverse\_iterator<T> &operator=(reverse\_iterator<T> it)

{

this->curr=it.curr;

return \*this;

}

reverse\_iterator<T> operator++()//++curr

{

curr--;

return curr;

}

reverse\_iterator<T> operator++(int)//++curr

{

reverse\_iterator<T> tmp=curr;

curr--;

return tmp;

}

T &operator\*() {return \*curr;}

bool operator!=(reverse\_iterator<T> t) {return curr!=t.curr;}

};

template<class T>

class Vector {

private:

unsigned int mySize; //so phan tu vector

unsigned int myCapacity; // kich thuoc bo nho

T \* buffer; //mang chua cac phan tu vector

public:

typedef T \* iterator;

iterator begin() { return buffer; } //vi tri bat dau bo lap xuoi

iterator end() { return buffer + mySize; } // vtri ket thuc bo lap xuoi

typedef reverse\_iterator<T> reverse\_it;

reverse\_it rbegin() {return reverse\_it(buffer+mySize-1);} // vi tri bat dau bo lap nguoc

reverse\_it rend() {return reverse\_it(buffer-1);} // vi tri ket thuc bo lap nguoc

Vector() : buffer(0), mySize(0), myCapacity(0) {}

~Vector() { //ham huy

delete [ ] buffer; //xoa bo nho dem

}

T& back() const { // tra ve gia tri cuoi

return buffer[mySize-1];

}

T& front() const { // tra ve gia tri dau

return buffer[0];

}

int capacity() { //Ham lay dung luong bo nho dem

return myCapacity;

}

unsigned int size() { //Ham lay so phan tu cua vector

return mySize;

}

bool empty() { //Ham kiem tra vector co rong hay khong

return mySize == 0;

}

void pop\_back() { //Xoa phan tu o cuoi vector

mySize--;

}

void reserve(unsigned int); // mo rong vector

void push\_back(T); // them vao cuoi

T &operator [](unsigned int index) { // nap chong toan tu []

return buffer[index];

}

Vector &operator=(Vector<T> V) //Nap chong toan tu = (gan hai vector)

{

this->mySize=V.mySize;

this->myCapacity=V.myCapacity;

if(myCapacity) {//Neu dung luong cua vector > 0

this->buff=new T[myCapacity]; //Khoi tao bo nho chua cac phan tu

for(int i=0;i<mySize;i++) //Sao chep cac phan tu

this->buffer[i]=V.buffer[i];

}

else this->buffer=0;

return \*this;

}

void insert(unsigned int , T); // ham chen vao vi tri

void erase(unsigned int,T);

void erase(iterator); // ham xoa qua vi tri bo lap

};

// MEMBER FUNCTIONS

template<class T>

void Vector<T>::reserve(unsigned int newCap) { // mo rong vector

if(buffer == 0) {

mySize = 0;

myCapacity = 0;

}

if(newCap <= myCapacity) { //dung luong moi < dung luong hien tai

return;

}

myCapacity = newCap; // mo rong dung luong

T \* newBuffer = new T[myCapacity]; //Khoi tao bo nho moi lon hon

for(int i=0;i<mySize;i++) //Luu lai cac phan tu da co trong vector

newBuffer[i]=buffer[i];

delete [ ] buffer; ///Xoa bo nho dem cu

buffer = newBuffer; //Tro lai bo nho dem

}

template<class T>

void Vector<T>::push\_back(T value) { //them phan tu vao cuoi vector

if(mySize >= myCapacity) {

reserve(myCapacity\*2); // het dung luowng thi mo rong

}

buffer[mySize++] = value; // them phan tu

}

template<class T>

void Vector<T> ::insert(unsigned int k, T x) { // chen x vao vi tri k

if (myCapacity ==mySize) reserve(myCapacity\*2); // het dung luowng thi mo rong

for (int i=mySize-1;i>=k;i--){ // dich cac phan tu ve cuoi vector

buffer[i+1] = buffer[i];

}

buffer[k]=x; //gan vi tri x vao k

mySize++; //tang so phan tu vector

}

template<class T>

void Vector<T> :: erase(unsigned int k,T o) { // xoa phan tu vi tri k

if(k>mySize-1) return ;

o = buffer[k];

int r=k;

while (r < mySize -1){

buffer[r]=buffer[r+1];

k++;

}

mySize--;

}

int main () {

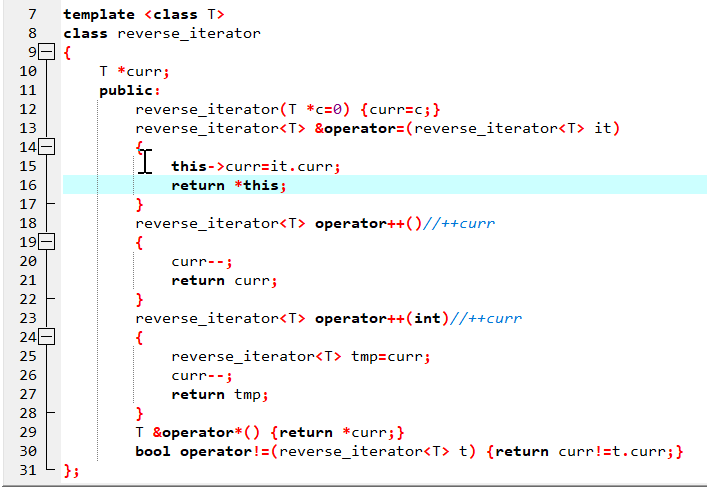
}

## ***2.Ý tưởng và mô tả thuật toán( thuộc tính ,phương thức các lớp và thời gian chạy phương thức)***

Sử dụng hàm mẫu T để khai báo con trỏ \*curr , xây dựng class reverse\_iterator để thực hiện công viêc khai báo vector có bộ lặp ngược, trong public bao gồm các phương thức:

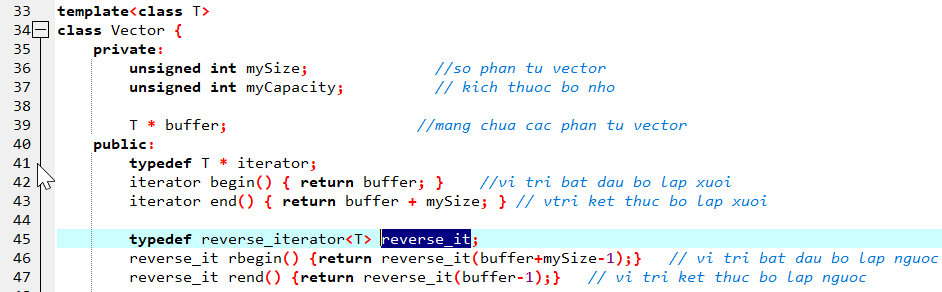
* Hàm khai báo không đối
* Xây dựng toán tử gán =
* Xây dựng toán tử ++curr có đối và không có đối
* Xây đựng toán tử !=
* Xây dựng toán tử trả về iterator \*curr

Cụ thể :



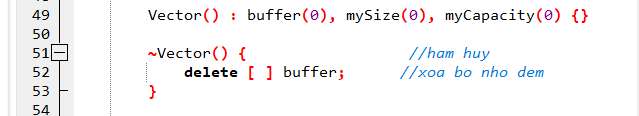
Ứng dụng hầm mẫu xâu dựng class vector có các thuộc tính . phương thức :

* Private : có các thông tin để kiểu số nguyên dương gồm số phần tử vector ,kích thước bộ nhớ và mảng chứa
* Public : khai báo cấu trúc con trỏ \* iterator và reverse\_it , tiến hành khai báo các vị trí đầu và cuối của bộ lặp xuôi ( begin, end) và bộ lặp ngược (rbegin, rend)

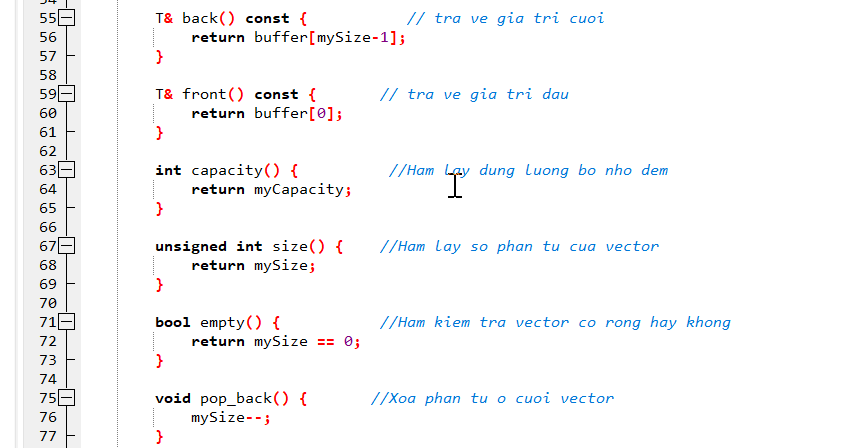


Trong public : xây dựng các phương thức sau :

* Hàm tạo vector không đối
* Hàm hủy vector



* Hàm trả về giá trị đầu front() và cuối back()
* Hàm lấy dung lượng bộ nhớ capacity()
* Hàm trả về số phần tử size()
* Hàm kiểm tra độ rỗng empty()
* Hàm xóa phần tử cuối popback()



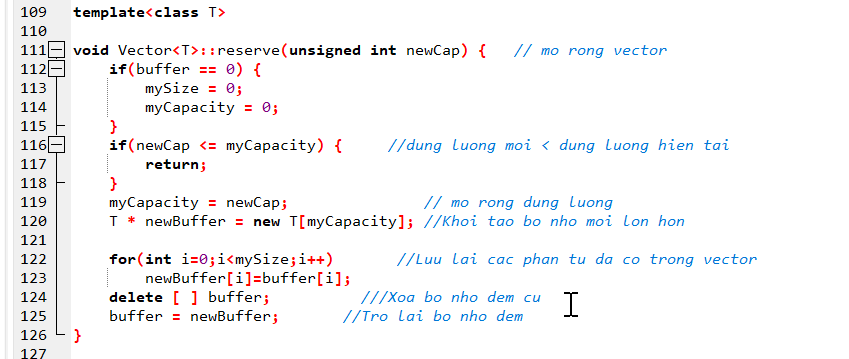
* Hàm mở rộng vector : reserve(unsigned int newCap) : đầu tiên , khởi tạo các thuộc tính private = 0, hàm rỗng khi bộ nhớ mới < bộ nhớ cũ, còn không thì trả gán myCapacity = newCap (cũ gán mới) .

+ Khởi tạo mảng mới lưu có bộ nhớ mới lớn hơn newBuffer

+ Lặp qua các phần tử mảng cũ , lưu nó vào mảng mới, xóa mảng cũ và tạo bộ nhớ đệm cho mảng mới

* Thời gian chạy: dòng 112,113,114 là 1 ; 116 là 2; 117 là 1; 119 là 2; 122 là 2+n ;123 là 2n;124 là 1; 125 là 2; tổng 3n+1

Cụ thể :



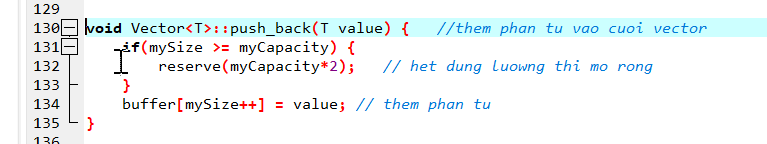
* Hàm thêm phần tử vào cuối push\_back() , khai báo 1 giá trị thêm value

+ Đối chiếu số phần tử vs bộ nhớ , nếu > = thì tăng bộ nhớ gấp đôi

+ Tiến hành thêm

* Xét thời gian chạy : dòng 131 là 2, dòng 132 = 3n+1 + 1; 134 là 2; tổng 3n+1

Cụ thể



* Hàm chèn phần tử insert(unsigned int k, T x) , khai báo giá trị x chèn vào vị trí k

+ Nếu số phần tử bằng bộ nhớ thì mở rộng bộ nhớ gấp 2 lần

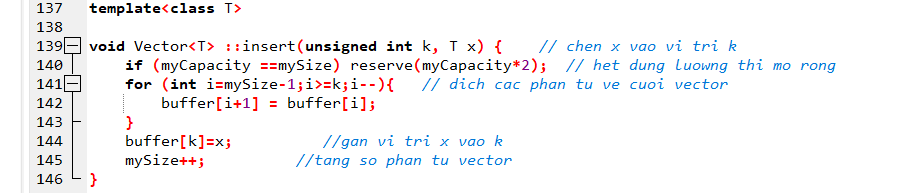
+ Dùng vòng lặp duyệt ngược từ n-1(mySize -1), chuyển các phần tử về cuối ,

+ Thoát khỏi vòng lặp, gắn vị trí của x vào phần tử thứ k, rồi tăng số phần tử mảng

* Thời gian chạy: dòng 140 là 3+2(3n+1); 141 là 2+n-k ;142 là 3(2+n-k);

144 là 3, 145 là 1; tổng là 9n+1

Cụ thể :



* Hàm xóa phần tử ở vị trí chỉ định erase(unsigned int k,T o)

+ Đầu tiên nếu vị trí nhập k chèn > n(mySize ) thì trả về rỗng

+ Gán phần tử o vào phần tử vị trí k

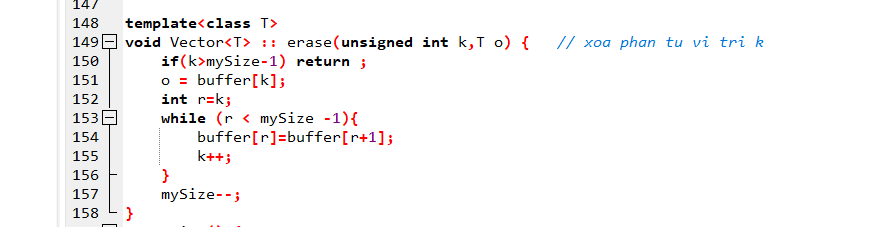
+ Gán ẩn thay thế r =k

+ Lặp While khi r<n-1, gán phần tử xóa bằng phần tử đứng sau nó, k++

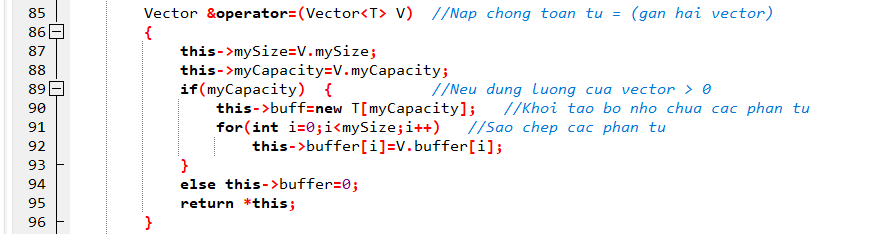
+ Thoát vòng lặp , giảm n(mySize ) đi 1đơn vị

* Thời gian chạy : dòng 150 là 2; 151 là k+1;152 là 2; 153 là n+1;154 là 3(n+1) . dòng 155 là n+1; 157 là 1; tổng là 3n+k+1

Cụ thể :



* Hàm nạp chồng toán tử =



# **C. Bài 2**

## ***1.Full sourses***

#include<iostream>

#include<string>

#include<vector>

#include<fstream>

#include<iomanip>

using namespace std;

class CAUHOI{

string cauhoi;

string dung;

string sai1, sai2, sai3;

public:

void nhap();

void xuat();

};

void CAUHOI::nhap() { 1

cin.ignore(); 2

getline(cin,cauhoi); 1

cout<<"nhap tra loi dung: ";cin.ignore(); 2

getline(cin,dung); 1

cout<<"nhap tra loi sai: ";cin.ignore(); 2

getline(cin,sai1); 1

cout<<"nhap tra loi sai: ";cin.ignore(); 2

getline(cin,sai2); 1

cout<<"nhap tra loi sai: ";cin.ignore(); 2

getline(cin,sai3); 1

}

void CAUHOI::xuat() {

cout<< " "<<cauhoi<<endl; 1

cout<<"A."<<dung<<endl; 1

cout<<"B."<<sai1<<endl; 1

cout<<"C."<<sai2<<endl; 1

cout<<"D."<<sai3<<endl; 1

}

int main() {

int n; cout << "nhap so cau hoi : ";cin>>n;

vector<CAUHOI> a(n);

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

cout<< "\nnhap cau hoi "<<i<<" : ";

a[i].nhap();

}

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

a[i].xuat(); return 0;

}

## ***2.Ý tưởng và mô tả thuật toán( thuộc tính ,phương thức các lớp và thời gian chạy phương thức)***

* Lớp có các thuộc tính private : 1 đáp án đúng và 3 đáp án sai được khai báo kiểu string
* Trong public có các phương thức nhập xuất cơ bản sử dụng cout , cin, cin.ignore( ) nhập xuất biến string
* Hàm xuất in ra màn hình câu hỏi và đáp án mà trong trường hợp nhỏ nhất là mặc định đáp án đúng là A
* Trong hàm main () ; ta khai báo n để nhập và test n câu hỏi , khai báo mảng vector để lưu danh sách n câu hỏi
* Dùng vòng for để nhập n câu hỏi và xuất n câu hỏi

## **D. Bài 3**

#include<iostream>

#include<string>

#include<vector>

#include<fstream>

#include<iomanip>

#include <ctime>

using namespace std;

typedef struct {

string deBai;

string dung;

string sai1,sai2,sai3;

} tracNghiem;

class DETHI{

private:

string dapAn;

vector<tracNghiem> cauHoi;

public:

void docFile(ifstream& FileCauHoi,string Filename);

void inDeThi();

void kiemtra();

};

void DETHI::docFile(ifstream& FileCauHoi, string Filename){

tracNghiem arrCauHoi;

FileCauHoi.open(Filename);

if (!FileCauHoi) {

cout <<"Error : can't not open !!";

exit(0);

}

while(!FileCauHoi.eof()) {

getline(FileCauHoi, arrCauHoi.deBai, '\n'); 1

getline(FileCauHoi, arrCauHoi.dung, '\n'); 1

getline(FileCauHoi, arrCauHoi.sai1, '\n'); 1

getline(FileCauHoi, arrCauHoi.sai2, '\n'); 1

getline(FileCauHoi, arrCauHoi.sai3, '\n'); 1

cauHoi.push\_back(arrCauHoi); 1

}

FileCauHoi.close(); 1

}

void DETHI::inDeThi() {

int kt[cauHoi.size()]; n

//srand(int( time(NULL)));

for(int i=0;i<cauHoi.size();i++){ n+2

srand(int( time(0))); n

kt[i]=rand()%24; 2n

if(kt[i]==0){ 2n

cout<< cauHoi[i].deBai<<endl; 2n

cout<<"A."<<cauHoi[i].dung<<endl; 2n

cout<<"B."<<cauHoi[i].sai1<<endl; 2n

cout<<"C."<<cauHoi[i].sai2<<endl; 2n

cout<<"D."<<cauHoi[i].sai3<<endl; 2n

dapAn='A'; 2n

}

if(kt[i]==1){ //tương tự

cout << cauHoi[i].deBai<<endl;

cout<<"A."<<cauHoi[i].dung<<endl;

cout<<"B."<<cauHoi[i].sai1<<endl;

cout<<"C."<<cauHoi[i].sai3<<endl;

cout<<"D."<<cauHoi[i].sai2<<endl;

dapAn='A';

}

if(kt[i]==2){

cout <<cauHoi[i].deBai<<endl;

cout<<"A."<<cauHoi[i].dung<<endl;

cout<<"B."<<cauHoi[i].sai2<<endl;

cout<<"C."<<cauHoi[i].sai1<<endl;

cout<<"D."<<cauHoi[i].sai3<<endl;

dapAn='A';

}

if(kt[i]==3){

cout << cauHoi[i].deBai<<endl;

cout<<"A."<<cauHoi[i].dung<<endl;

cout<<"B."<<cauHoi[i].sai2<<endl;

cout<<"C."<<cauHoi[i].sai3<<endl;

cout<<"D."<<cauHoi[i].sai1<<endl;

dapAn='A';

}

if(kt[i]==4){

cout << cauHoi[i].deBai<<endl;

cout<<"A."<<cauHoi[i].dung<<endl;

cout<<"B."<<cauHoi[i].sai3<<endl;

cout<<"C."<<cauHoi[i].sai2<<endl;

cout<<"D."<<cauHoi[i].sai1<<endl;

dapAn='A';

}

if(kt[i]==5){

cout << cauHoi[i].deBai<<endl;

cout<<"A."<<cauHoi[i].dung<<endl;

cout<<"B."<<cauHoi[i].sai3<<endl;

cout<<"C."<<cauHoi[i].sai1<<endl;

cout<<"D."<<cauHoi[i].sai2<<endl;

dapAn='A';

}

if(kt[i]==6){

cout << cauHoi[i].deBai<<endl;

cout<<"A."<<cauHoi[i].sai1<<endl;

cout<<"B."<<cauHoi[i].dung<<endl;

cout<<"C."<<cauHoi[i].sai2<<endl;

cout<<"D."<<cauHoi[i].sai3<<endl;

dapAn='B';

}

if(kt[i]==7){

cout << cauHoi[i].deBai<<endl;

cout<<"A."<<cauHoi[i].sai1<<endl;

cout<<"B."<<cauHoi[i].dung<<endl;

cout<<"C."<<cauHoi[i].sai3<<endl;

cout<<"D."<<cauHoi[i].sai2<<endl;

dapAn='B';

}

if(kt[i]==8){

cout << cauHoi[i].deBai<<endl;

cout<<"A."<<cauHoi[i].sai2<<endl;

cout<<"B."<<cauHoi[i].dung<<endl;

cout<<"C."<<cauHoi[i].sai1<<endl;

cout<<"D."<<cauHoi[i].sai3<<endl;

dapAn='B';

}

if(kt[i]==9){

cout << cauHoi[i].deBai<<endl;

cout<<"A."<<cauHoi[i].sai2<<endl;

cout<<"B."<<cauHoi[i].dung<<endl;

cout<<"C."<<cauHoi[i].sai3<<endl;

cout<<"D."<<cauHoi[i].sai1<<endl;

dapAn='B';

}

if(kt[i]==10){

cout << cauHoi[i].deBai<<endl;

cout<<"A."<<cauHoi[i].sai3<<endl;

cout<<"B."<<cauHoi[i].dung<<endl;

cout<<"C."<<cauHoi[i].sai1<<endl;

cout<<"D."<<cauHoi[i].sai2<<endl;

dapAn='B';

}

if(kt[i]==11){

cout << cauHoi[i].deBai<<endl;

cout<<"A."<<cauHoi[i].sai3<<endl;

cout<<"B."<<cauHoi[i].dung<<endl;

cout<<"C."<<cauHoi[i].sai2<<endl;

cout<<"D."<<cauHoi[i].sai1<<endl;

dapAn='B';

}

if(kt[i]==12){

cout << cauHoi[i].deBai<<endl;

cout<<"A."<<cauHoi[i].sai1<<endl;

cout<<"B."<<cauHoi[i].sai2<<endl;

cout<<"C."<<cauHoi[i].dung<<endl;

cout<<"D."<<cauHoi[i].sai3<<endl;

dapAn='C';

}

if(kt[i]==13){

cout << cauHoi[i].deBai<<endl;

cout<<"A."<<cauHoi[i].sai1<<endl;

cout<<"B."<<cauHoi[i].sai3<<endl;

cout<<"C."<<cauHoi[i].dung<<endl;

cout<<"D."<<cauHoi[i].sai2<<endl;

dapAn='C';

}

if(kt[i]==14){

cout << cauHoi[i].deBai<<endl;

cout<<"A."<<cauHoi[i].sai2<<endl;

cout<<"B."<<cauHoi[i].sai1<<endl;

cout<<"C."<<cauHoi[i].dung<<endl;

cout<<"D."<<cauHoi[i].sai3<<endl;

dapAn='C';

}

if(kt[i]==15){

cout << cauHoi[i].deBai<<endl;

cout<<"A."<<cauHoi[i].sai2<<endl;

cout<<"B."<<cauHoi[i].sai3<<endl;

cout<<"C."<<cauHoi[i].dung<<endl;

cout<<"D."<<cauHoi[i].sai1<<endl;

dapAn='C';

}

if(kt[i]==16){

cout << cauHoi[i].deBai<<endl;

cout<<"A."<<cauHoi[i].sai3<<endl;

cout<<"B."<<cauHoi[i].sai1<<endl;

cout<<"C."<<cauHoi[i].dung<<endl;

cout<<"D."<<cauHoi[i].sai2<<endl;

dapAn='C';

}

if(kt[i]==17){

cout << cauHoi[i].deBai<<endl;

cout<<"A."<<cauHoi[i].sai3<<endl;

cout<<"B."<<cauHoi[i].sai2<<endl;

cout<<"C."<<cauHoi[i].dung<<endl;

cout<<"D."<<cauHoi[i].sai1<<endl;

dapAn='C';

}

if(kt[i]==18){

cout << cauHoi[i].deBai<<endl;

cout<<"A."<<cauHoi[i].sai1<<endl;

cout<<"B."<<cauHoi[i].sai2<<endl;

cout<<"C."<<cauHoi[i].sai3<<endl;

cout<<"D."<<cauHoi[i].dung<<endl;

dapAn='D';

}

if(kt[i]==19){

cout << cauHoi[i].deBai<<endl;

cout<<"A."<<cauHoi[i].sai1<<endl;

cout<<"B."<<cauHoi[i].sai3<<endl;

cout<<"C."<<cauHoi[i].sai2<<endl;

cout<<"D."<<cauHoi[i].dung<<endl;

dapAn='D';

}

if(kt[i]==20){

cout << cauHoi[i].deBai<<endl;

cout<<"A."<<cauHoi[i].sai2<<endl;

cout<<"B."<<cauHoi[i].sai1<<endl;

cout<<"C."<<cauHoi[i].sai3<<endl;

cout<<"D."<<cauHoi[i].dung<<endl;

dapAn='D';

}

if(kt[i]==21){

cout << cauHoi[i].deBai<<endl;

cout<<"A."<<cauHoi[i].sai2<<endl;

cout<<"B."<<cauHoi[i].sai3<<endl;

cout<<"C."<<cauHoi[i].sai1<<endl;

cout<<"D."<<cauHoi[i].dung<<endl;

dapAn='D';

}

if(kt[i]==22){

cout << cauHoi[i].deBai<<endl;

cout<<"A."<<cauHoi[i].sai3<<endl;

cout<<"B."<<cauHoi[i].sai1<<endl;

cout<<"C."<<cauHoi[i].sai2<<endl;

cout<<"D."<<cauHoi[i].dung<<endl;

dapAn='D';

}

if(kt[i]==23){

cout << cauHoi[i].deBai<<endl;

cout<<"A."<<cauHoi[i].sai3<<endl;

cout<<"B."<<cauHoi[i].sai2<<endl;

cout<<"C."<<cauHoi[i].sai1<<endl;

cout<<"D."<<cauHoi[i].dung<<endl;

dapAn='D';

}

}

}

void DETHI::kiemtra() { 1

int diem=0; 2

string cauTrl;

cout << "\nNhap cau tra loi cua ban (A/B/C or D): \n"; 1

for(int i=0;i< cauHoi.size();i++) { n+2

cout<<"cau "<< i+1<<" : "; n

cin>>cauTrl; n

if (dapAn == cauTrl) diem++; 4n

}

cout << "\nDiem cua ban la : "<< diem<<"/"<<cauHoi.size(); n

}

int main() {

DETHI deThi;

ifstream readDeThi;

deThi.docFile(readDeThi , "dethi.txt");

string k;

do {

cout <<"\nban co muon kiem tra (1/0) : ";

cin >>k;

deThi.inDeThi();

deThi.kiemtra();

} while (k=="1");

return 0;

}

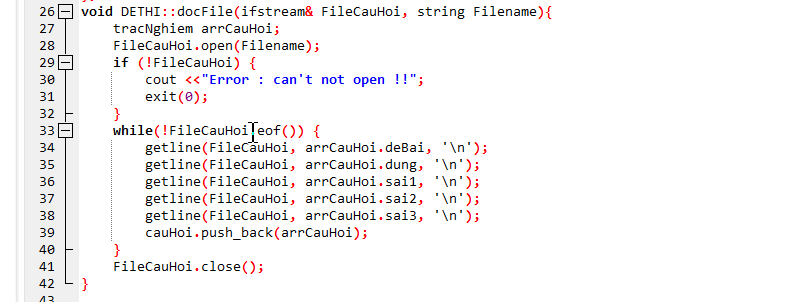
## ***2.Ý tưởng và mô tả thuật toán( thuộc tính ,phương thức các lớp )***

* Khai báo hàm cấu trúc câu hỏi trắc nghiệm gồm đề câu hỏi ,đáp án đúng và 3 đáp án sai, và cả đề bài của câu hỏi
* Xây dưng lớp đề thi bao gồm các thuộc tính:

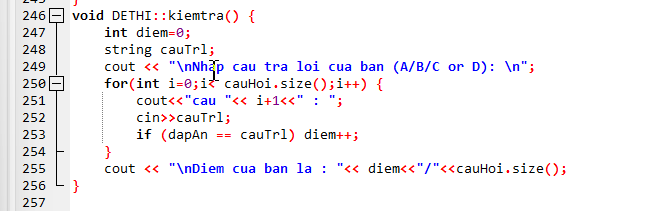
+ Private :khai báo đáp án kiểu string và vector lưu trữ cấu trúc trắc nghiệm

+ Public :thực hiện các phương thức :

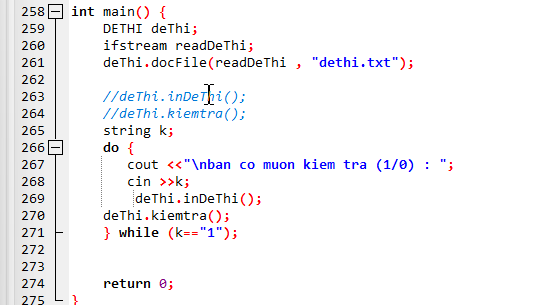
* Đọc file : từ văn bản định dạng .txt : mở file bằng lệnh open và gắn theo đó bằng các tên filename .Đầu tiên ta kiểm tra xem file có tồn tại hay ko bằng lệnh if. Sau đó, nếu file còn lưu trữ, ta thực hiện lệnh đọc từng dòng trong file bằng hàm getline ( cứ gặp \n là kết thúc lệnh)



* In ra đề thi: dùng hoán vị cho 4 đáp án bằng random() để tạo ra các câu hỏi sáo trộn vị trí đáp án trả lời. Ta khai biến kiểm tra có số phần tử = n câu hỏi, dùng vòng lặp qua n câu hỏi random theo thời gian để lấy cách xuất hiện các câu trả lời. đồng thời trong mỗi cách random, ta gán giá trị đáp án đúng cho từng cách xuất hiện là các chữ cái A/B/C hoặc D ( code như bên trên)
* Kiểm tra : khai báo biến string cauTrl để nhập câu trả lời, và diem để tính điểm của bài thi , lặp qua lần lượt các câu hỏi, mỗi câu hỏi bạn sẽ điền đáp án theo mẫu quy định, nếu trừng với mẫu dapAn thì diem++, sau đó in ra điểm trên màn hình



* Trong main (), ta thực hiện các hàm trên, sử dụng thêm vòng lặp do While để có thể in đề thi và thực hiện kiểm tra nhiều lần



# **E. Tài liệu tham khảo**

[1**] Đỗ Xuân Lôi**: "Cấu trúc dữ liệu". Đại học Bách khoa Hà Nội xuất bản

- 1976.

[2] **Đỗ Xuân Lôi**: "Sắp xếp và tìm kiếm dữ liệu trên máy tính điện tử". Đại

học Bách khoa Hà Nội xuất bản - 1980.

[3] **Nguyễn Xuân Huy:** "Thuật toán". Nhà xuất bản Thống kê - 1988.

[4] **Đoàn Nguyên Hải - Nguyễn Trung Trực - Nguyễn Anh Dũng**: "Lập trình căn bản". Trung tâm điện toán - Đại học Bách khoa TP. HCM. 1991.

[5] **Jacques Arsac**: "Nhập môn lập trình". Bản dịch của **Nguyễn Chí Công -** **Đinh Văn Phong - Trần Ngọc Trí**. Trung tâm hệ thống thông tin ISC - 1991.

[6] **J. Courtin, L. Kowarski**: "Initiation à l'algorithmique et aux structures de données". **Nhà xuất bản DUNOD - 1990.**

(7) **Larry N. Hoff, Sanford hesstma**: "Lập trình nâng cao bằng PASCAL với cấu trúc dữ liệu". **Bản dịch của Lê Minh Trung - Công ty**

**SCITEC xuất bản 1991**.

[8] **Donald Knuth**: "The art of computer programming": vol 1: Fundamental algorithms; vol 3: Sorting and searching. **Addision Wesley Publishing Company 1973.**

[9] **Niklaus Wirth**: "Algorithms + data structures = programs". **Prentice- Hall INC - 1976.**

[10] **Mark Elson**: "Data Structures". Science Research Associates INC - 1975.

[11] **Ellis Horowitz, Sartaj Sahani**: "Fundamentals of data structures". Computer Science Press INC- 1976.

[12] **Robert J. Baron, Linde G. Shapiro**: "Data structures and their implementation". **Van Vostrand Reingold company - 1980**.

[13] **A.V. Aho, J.E. Hopcroft, J.D. Ulmann**: "Data structures and algorithms". **Addison Wesley - 1983**.

[14] **Jean Paul Tremblay, Paul G. Sorenson**: "An introduction to data structures with applications". **McGraw-Hill INC - 1984**.

[15] **Paul Helman, Robert Veroff**: "Intermediate problem solving and data structures". The Benjamin/cummings publishing company**. INC - 1986**.