**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**

Giảng viên hướng dẫn: ThS. Phạm Xuân Tích

Sinh viên thực hiện: Nguyễn Việt Anh\_CNTT1\_201200016

**Hà Nội, tháng 12 năm 2021**

**MỤC LỤC**

1. Giới thiệu đề tài

1. Phân tích bài toán Xác định các yêu cầu của bài toán, xác định các lớp, các thuộc tính, các phương thức của lớp. Mô tả chức năng của từng lớp, từng phương thức. Vẽ sơ đồ khối của những phương thức có chứa các cấu trúc if, lặp, đánh giá thời gian chạy của từng phương thức.
2. Cài đặt các lớp và hàm main bằng C++
3. Tài liệu tham khảo

I. Đề bài

Một công ty điện thoại cần xây dựng một chương trình tính tiền điện thoại cho các khách hàng. Hiện tại, công ty có lưu trữ 2 tệp tin, một tệp về khách hàng, một tệp về các cuộc điện thoại đã gọi của khách hàng.

1. Tệp khách hàng có tên ***khachhang.txt***. Mỗi dòng của tệp này gồm có:

*Tên khách hàng; số điện thoại*

- Tên của khách hàng: tên là một xâu ký tự, độ dài không vượt quá 25 và kết thúc bởi dấy “;”.

- Số điện thoại của khách hàng gồm 10 chữ số bắt đầu bằng chữ số 0

Ví dụ:

Nguyen Anh Tuấn; 0987654233

Le Nhat Anh; 0967456321

…

2. Tệp lưu trữ các cuộc điện thoại có tên ***cuocgoi.txt***. Mỗi dòng có những thông tin về một cuộc điện thoại như sau:

*Số điện thoại; Số phút; Thời điểm bắt đầu gọi; Ngày gọi (dd/mm/yyyy); Vùng*

(Vùng: nội hạt, lân cận, xa và rất xa, được viết tắt là: NH, LC, X, RX).  
Ví dụ:

0987654233; 4; 8h23; 14/010/2018; NH

0987654233; 6; 15h10; 19/10/2018; LC

0967456321; 1; 23h05; 15/10/2018; RX

….

Hãy sử dụng cấu trúc dữ liệu thích hợp viết chương trình tính tiền điện thoại theo yêu cầu sau:

- Đọc thông tin trong tệp *khachhang.txt* và ***cuocgoi.txt***, tính tiền cho từng khách hàng và ghi ra tệp ketqua.txt, mỗi dòng có các thông tin sau:

*Tên KH; số đt; Số tiền, số cuộc gọi NH, số gọi LC, số cuộc X, số cuộc gọi RX.*

***Cách tính tiền cho mỗi cuộc gọi như sau:***

*Tiền = Giá cơ bản \* Số phút \* Hệ số miền.*

*Giá cơ bản là 1.100 đồng 1 phút.*

*Hệ số miền đối với nội hạt là 1, với lân cận là 2, với xa là 3, với rất xa là 4.  
Đối với các cuộc gọi bắt đầu từ 23h00 đến 5h00 các ngày trong tuần và ngày Thứ Bảy, Chủ nhật thì được giảm gái 30%.*

Lưu ý: trong danh sách cuộc gọi mỗi khách có thể có nhiều cuộc gọi hoặc không có cuộc gọi nào.

II. Phân tích bài toán

**1, Yêu cầu của bài toán**

Sử dụng cấu trúc dữ liệu thích hợp viết chương trình tính tiền điện thoại theo yêu cầu sau:

- Đọc thông tin trong tệp ***khachhang.txt*** và ***cuocgoi.txt***, tính tiền cho từng khách hàng và ghi ra tệp ketqua.txt, mỗi dòng có các thông tin sau:

*Tên KH; số đt; Số tiền, số cuộc gọi NH, số gọi LC, số cuộc X, số cuộc gọi RX.*

2, Các lớp, các thuộc tính, các phương thức của lớp

- Lớp cuộc gọi (Cuocgoi): Lớp dùng để khai báo những thuộc tính của cuộc gọi: số điện thoại(sdt), số phút gọi(sophut), thời điểm bắt đầu gọi(thoidiembatdau), ngày gọi(tgian), vùng(vung).

- Doc\_file\_cg

- DocFile2

- getsdt, getsophut, gettimebatdau, gettime, getvung

-Lớp khách hàng (Khachhang): Lớp dùng để khai báo những thuộc tính của khách hàng: họ và tên (hoten), số điện thoại (sdt).

- Doc\_File\_kh.

- DocFile1.

-getname, getsodienthoai.

-Các phương thức khác:

-nhuan

-SoNgay

-TinhThu

-gio

-TinhTong

-Demso

-Out

-Out\_file

3, Mô tả chức năng của từng lớp, từng phương thức, từng hàm

- Lớp cuộc gọi (Cuocgoi): Lớp dùng để khai báo những thuộc tính của cuộc gọi: số điện thoại(sdt), số phút gọi(sophut), thời điểm bắt đầu gọi(thoidiembatdau), ngày gọi(tgian), vùng(vung).

-Lớp khách hàng (Khachhang): Lớp dùng để khai báo những thuộc tính của khách hàng: họ và tên (hoten), số điện thoại (sdt).

- Doc\_file\_cg: đọc thông tin 1 cuộc gọi từ file cuocgoi.txt. Thời gian chạy O(1)

- Doc\_File\_kh: đọc thông tin 1 khách hàng từ file khachhang.txt. Thời gian chạy O(1)

- DocFile2: đọc toàn bộ dữ liệu trong file cuocgoi.txt, sử dụng cấu trúc dữ liệu vector để lưu thông tin danh sách cuộc gọi. Thời gian chạy O(n).

- DocFile1: đọc toàn bộ dữ liệu trong file khachhang.txt, sử dụng cấu trúc dữ liệu vector để lưu thông tin danh sách khách hàng. Thời gian chạy O(n).

-Out: Ghi dữ liệu từng khách hàng (*Tên KH; số đt; Số tiền, số cuộc gọi NH, số gọi LC, số cuộc X, số cuộc gọi RX)* vào file *ketqua.txt*). Thời gian chạy O(1)

-Out\_file: Ghi dữ liệu toàn bộ khách hàng (*Tên KH; số đt; Số tiền, số cuộc gọi NH, số gọi LC, số cuộc X, số cuộc gọi RX)* vào file *ketqua.txt*).Thời gian chạy O(n)

-int nhuan(int year): xác định năm nhuận. Năm nhuận là năm mà chia hết cho 400 hoặc vừa chia hết cho 4 vừa không chia hết cho 100. Hàm trả về 0 hoặc 1. Nếu 1 thì năm đó là năm nhuận, 0 thì năm đó không phải là năm nhuận. Thời gian chạy O(1).

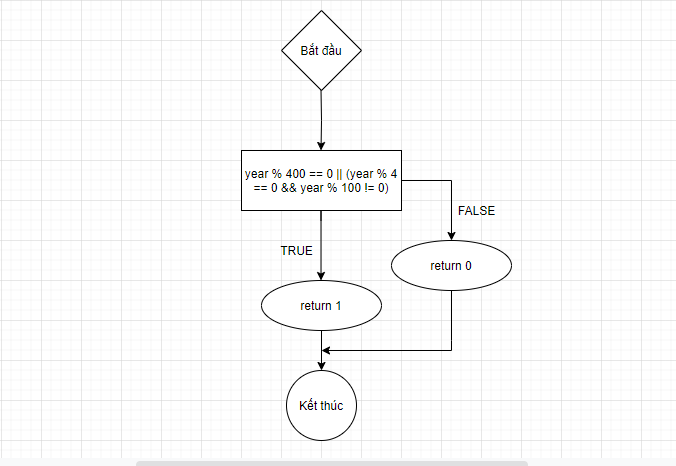
int nhuan(int year){

if (year % 400 == 0 || (year % 4 == 0 && year % 100 != 0))

return 1;

return 0;

}



-int SoNgay(): xác định tổng số ngày từ đầu năm đến ngày gọi. Ví dụ ngày gọi là 01/03/2021 thì tổng số ngày là 31+ 28 + 1 = 60. (vì 2021 không phải là năm nhuận nên tháng 2 chỉ có 28 ngày). Hàm trả về tổng số ngày từ đầu năm đến ngày gọi. Thời gian chạy O(n).

int SoNgay(int day,int month,int year){

int t[]={0,31,28,31,30,31,30,31,31,30,31,30,31};

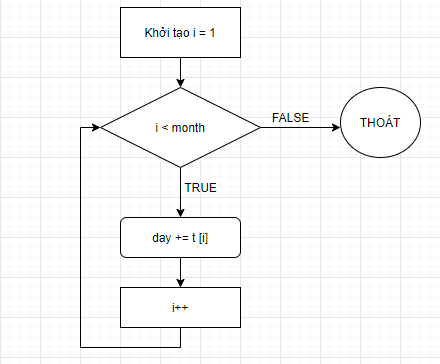
for(int i=1;i<month;i++)

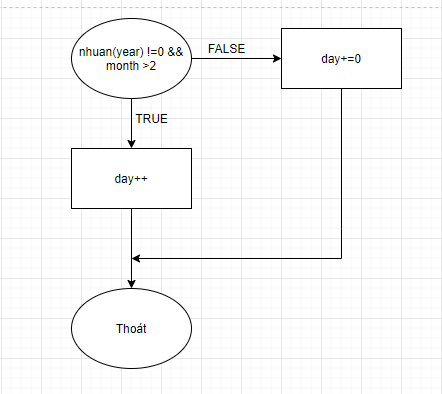
day+=t[i];

if(nhuan(year)&&month>2)day++;

return day;

}





-int TinhThu(string s): xác định chính xác ngày tháng năm gọi là thứ mấy. Công thức tính thứ:

S = (X - 1)/1 + ((X - 1)/4 - (X - 1)/100) + (X -1)/400 + C;

X: là năm gọi

C: tổng số ngày từ đầu năm đến ngày gọi.

Ta lấy S % 7

* S % 7 = 0 🡪 Chủ nhật.
* S % 7 = 1 🡪 Thứ hai.
* S % 7 = 2 🡪 Thứ ba.
* S % 7 = 3 🡪 Thứ tư.
* S % 7 = 4 🡪 Thứ năm.
* S % 7 = 5 🡪 Thứ sáu.
* S % 7 = 6 🡪 Thứ bảy.

Vì ngày gọi được lưu ở dạng string dd/mm/yyyy trong file cuocgoi.txt nên cần phải tách lấy ngày, tháng, năm và chuyển về kiểu số nguyên. Hàm trả về giá trị trong khoảng từ [0-6] tương ứng với từ [Chủ nhật-Thứ 2…-Thứ bảy]. Thời gian chạy O(1).

int TinhThu(string s){

int pos = s.find("/"); // tim ki tu "/" dau tien

int day = atoi(s.substr(0,pos).c\_str()); // lay chuoi truoc "/" dau tien, chuyen ve dang so.

s = s.substr(pos+1); // bo phan dau va ki tu "/" dau

pos = s.find("/"); // tim ki tu "/"

int month= atoi(s.substr(0,pos).c\_str()); // lay chuoi truoc "/" va chuyen ve dang so.

int year = atoi(s.substr(pos+1).c\_str()); // lay chuoi sau "/" va chuyen ve dang so.

int tongngay=SoNgay(day,month,year);

int sum=(year - 1)/1 + ((year - 1)/4 - (year - 1)/100) + (year -1)/400 + tongngay;

int thu =sum%7;

return thu;

}

-int gio(string s): xác định giờ của cuộc gọi.Hàm trả về sô nguyên tương ứng với giờ của cuộc gọi. Thời gian chạy O(1).

int gio(string s){

int pos = s.find("h"); // tim ki tu "h".

int t= atoi(s.substr(0,pos).c\_str()); // lay chuoi truoc ki tu "h" va chuyen ve dang so.

return t;

}

-TinhTong: Tính số tiền phải trả của một khách hàng.

**Cách tính tiền cho mỗi cuộc gọi như sau:**

*Tiền = Giá cơ bản \* Số phút \* Hệ số miền.*

Giá cơ bản là 1.100 đồng 1 phút.

Hệ số miền đối với nội hạt là 1, với lân cận là 2, với xa là 3, với rất xa là 4.  
Đối với các cuộc gọi bắt đầu từ 23h00 đến 5h00 các ngày trong tuần và ngày Thứ Bảy, Chủ nhật thì được giảm gái 30%. Thời gian chạy O().

void TinhTong(vector<Cuocgoi> DS\_CG,vector<Khachhang> DS\_KH,int sokh,int socg,int \*tienTong){

int \*Tien;

Tien=new int [socg];

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

Tien[i]=0;

}

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

if(DS\_CG[i].getvung() ==" NH") Tien[i]=1100\*DS\_CG[i].getsophut()\*1;

if(DS\_CG[i].getvung()==" LC") Tien[i]=1100\*DS\_CG[i].getsophut()\*2;

if(DS\_CG[i].getvung()==" X") Tien[i]=1100\*DS\_CG[i].getsophut()\*3;

if(DS\_CG[i].getvung()==" RX") Tien[i]=1100\*DS\_CG[i].getsophut()\*4;

}

int \*Thu;

Thu=new int[socg+1];

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

Thu[i]=TinhThu(DS\_CG[i].gettime());

}

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

if(gio(DS\_CG[i].gettimebatdau())==23|| gio(DS\_CG[i].gettimebatdau()) <=5 || Thu[i]==0||Thu[i]==6)

Tien[i]=Tien[i]\*0.7;

}

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

tienTong[i]=0;

}

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

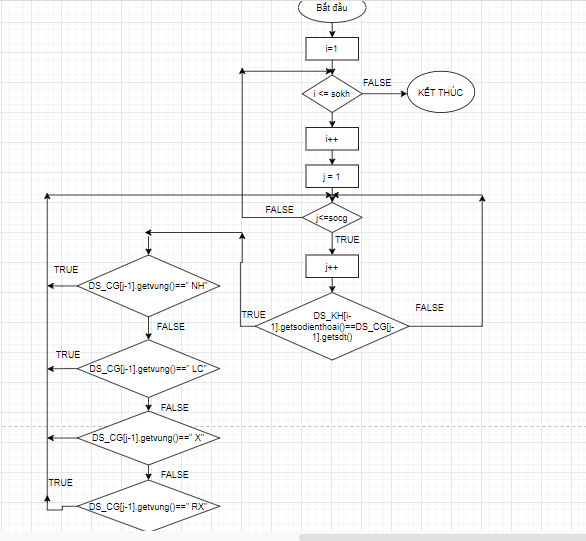
for(int j=0;j<socg;j++)

if(DS\_CG[j].getsdt()==DS\_KH[i].getsodienthoai()) tienTong[i]+=Tien[j];

}

}

-Demso: đếm số cuộc gọi NH, LC, X, RX của từng khách hàng. Thời gian chạy O().



void Demso(vector<Cuocgoi> DS\_CG,vector<Khachhang> DS\_KH,int sokh,int socg,int \*\*dem)

{

for(int i=1;i<=sokh;i++)

for(int j=1;j<=4;j++)

{

dem[i][j]=0;

}

for(int i=1;i<=sokh;i++)

for(int j=1;j<=socg;j++)

{

if(DS\_KH[i-1].getsodienthoai()==DS\_CG[j-1].getsdt())

{

if(DS\_CG[j-1].getvung()==" NH") dem[i][1]++;

if(DS\_CG[j-1].getvung()==" LC") dem[i][2]++;

if(DS\_CG[j-1].getvung()==" X") dem[i][3]++;

if(DS\_CG[j-1].getvung()==" RX") dem[i][4]++;

}

}

}

3. Cài đặt các lớp và hàm main bằng C++

#include<bits/stdc++.h>

using namespace std;

class Cuocgoi{

long int sdt;

int sophut;

string thoidiembatdau;

string tgian;

string vung;

public:

friend void Doc\_file\_cg(ifstream &filein,Cuocgoi &cg){

char c;

filein>>cg.sdt;

filein>>c; // doc dau ';'

filein>>cg.sophut;

filein>>c; // doc dau ';'

getline(filein, cg.thoidiembatdau,';');

getline(filein, cg.tgian,';');

getline(filein, cg.vung);

}

friend void DocFile2(ifstream &filein,vector<Cuocgoi> &DS\_CG,int &socg){

while(filein.eof()==false){

Cuocgoi cg;

Doc\_file\_cg(filein,cg);

DS\_CG.push\_back(cg);

}

socg=DS\_CG.size()-1;

}

long int getsdt(){

return sdt;

}

int getsophut(){

return sophut;

}

string gettimebatdau(){

return thoidiembatdau;

}

string gettime(){

return tgian;

}

string getvung(){

return vung;

}

};

class Khachhang{

string hoten;

long int sdt;

public:

friend void Doc\_File\_kh(ifstream &filein,Khachhang &kh){

getline(filein, kh.hoten,';');

filein>>kh.sdt;

// tao bien tam de doc ki tu xuong dong o cuoi dong.

string temp;

getline(filein, temp);

}

friend void DocFile1(ifstream &filein,vector<Khachhang> &DS\_KH,int &sokh){

while(filein.eof()==false){

Khachhang kh;

Doc\_File\_kh(filein,kh);

DS\_KH.push\_back(kh);

}

sokh=DS\_KH.size()-1;

}

string getname(){

return hoten;

}

long int getsodienthoai(){

return sdt;

}

};

void Out(ofstream &fileout,Khachhang KH,int tienTong,int d1,int d2,int d3,int d4){

fileout<<KH.getname()<<";"<<"0"<<KH.getsodienthoai()<<";"<<tienTong<<","<<d1<<","<<d2<<","<<d3<<","<<d4<<"\n";

}

void Out\_file(ofstream &fileout,vector<Khachhang> DS\_KH,int nkh,int \*Tong,int \*\*dem){

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

Out(fileout,DS\_KH[i],Tong[i],dem[i+1][1],dem[i+1][2],dem[i+1][3],dem[i+1][4]);

}

}

// ham kiem tra nam nhuan

int nhuan(int year){

if (year % 400 == 0)

return 1;

if (year % 4 == 0 && year % 100 != 0)

return 1;

return 0;

}

// ham tinh tong so ngay

int SoNgay(int day,int month,int year){

int t[]={0,31,28,31,30,31,30,31,31,30,31,30,31};

for(int i=1;i<month;i++)

day+=t[i];

if(nhuan(year)&&month>2)day++;

return day;

}

// ham tinh thu

int TinhThu(string s){

int pos = s.find("/"); // tim ki tu "/" dau tien

int day = atoi(s.substr(0,pos).c\_str()); // lay chuoi truoc "/" dau tien, chuyen ve dang so.

s = s.substr(pos+1); // bo phan dau va ki tu "/" dau

pos = s.find("/"); // tim ki tu "/"

int month= atoi(s.substr(0,pos).c\_str()); // lay chuoi truoc "/" va chuyen ve dang so.

int year = atoi(s.substr(pos+1).c\_str()); // lay chuoi sau "/" va chuyen ve dang so.

int tongngay=SoNgay(day,month,year);

int sum=(year - 1)/1 + ((year - 1)/4 - (year - 1)/100) + (year -1)/400 + tongngay;

int thu =sum%7;

return thu;

}

// ham lay gio

int gio(string s){

int pos = s.find("h"); // tim ki tu "h".

int t= atoi(s.substr(0,pos).c\_str()); // lay chuoi truoc ki tu "h" va chuyen ve dang so.

return t;

}

void TinhTong(vector<Cuocgoi> DS\_CG,vector<Khachhang> DS\_KH,int sokh,int socg,int \*tienTong){

int \*Tien;

Tien=new int [socg];

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

Tien[i]=0;

}

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

if(DS\_CG[i].getvung() ==" NH") Tien[i]=1100\*DS\_CG[i].getsophut()\*1;

if(DS\_CG[i].getvung()==" LC") Tien[i]=1100\*DS\_CG[i].getsophut()\*2;

if(DS\_CG[i].getvung()==" X") Tien[i]=1100\*DS\_CG[i].getsophut()\*3;

if(DS\_CG[i].getvung()==" RX") Tien[i]=1100\*DS\_CG[i].getsophut()\*4;

}

int \*Thu;

Thu=new int[socg+1];

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

Thu[i]=TinhThu(DS\_CG[i].gettime());

}

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

if(gio(DS\_CG[i].gettimebatdau())==23|| gio(DS\_CG[i].gettimebatdau()) <=5 || Thu[i]==0||Thu[i]==6)

Tien[i]=Tien[i]\*0.7;

}

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

tienTong[i]=0;

}

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

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

if(DS\_CG[j].getsdt()==DS\_KH[i].getsodienthoai()) tienTong[i]+=Tien[j];

}

}

void Demso(vector<Cuocgoi> DS\_CG, vector<Khachhang> DS\_KH, int sokh, int socg, int \*\*dem)

{

for(int i=1;i<=sokh;i++)

for(int j=1;j<=4;j++)

{

dem[i][j]=0;

}

for(int i=1;i<=sokh;i++)

for(int j=1;j<=socg;j++)

{

if(DS\_KH[i-1].getsodienthoai()==DS\_CG[j-1].getsdt())

{

if(DS\_CG[j-1].getvung()==" NH") dem[i][1]++;

if(DS\_CG[j-1].getvung()==" LC") dem[i][2]++;

if(DS\_CG[j-1].getvung()==" X") dem[i][3]++;

if(DS\_CG[j-1].getvung()==" RX") dem[i][4]++;

}

}

}

int main () {

ios\_base::sync\_with\_stdio(0);cin.tie(NULL);cout.tie(NULL);

int sokh,socg;

ifstream filein1;

ifstream filein2;

filein1.open("khachhang.txt",ios\_base::in);

vector<Khachhang> DS\_KH;

DocFile1(filein1,DS\_KH,sokh);

filein1.close();

filein2.open("cuocgoi.txt",ios\_base::in);

vector<Cuocgoi> DS\_CG;

DocFile2(filein2,DS\_CG,socg);

filein2.close();

int \*TienTong;

TienTong=new int [sokh];

TinhTong(DS\_CG,DS\_KH,sokh,socg,TienTong);

int \*\*dem;

dem=new int \*[sokh];

for(int i=1;i<=sokh;i++)

dem[i]=new int[4];

Demso(DS\_CG,DS\_KH,sokh,socg,dem);

ofstream fileout;

fileout.open("ketqua1.txt",ios\_base::out);

Out\_file(fileout,DS\_KH,sokh,TienTong,dem);

fileout.close();

system("pause");

return 0;

}

4. Tài liệu tham khảo

* Trang web: <https://www.cplusplus.com>
* Tài liệu nhập xuất từ file trên internet.
* Bài giảng của thầy Phạm Xuân Tích.