#include <iostream>

#include <string>

#include <fstream>

#include <sstream>

#include <regex>

using namespace std;

class QuanCo{

public:

int hang;// quan co nam o hang nao?

int cot; // quan co nam o cot nao?

bool trang;// co phai mau trang hay khong?

string ten;//ten quan co

//Quan co co the di chuyen den vi tri moi khong tren co so ban co hien tai? Tra ve true neu di chuyen duoc va false neu khong di chuyen duoc.

virtual bool move(int hangmoi, int cotmoi, QuanCo\* \*\*banco) = 0;

};

class Vua: public QuanCo{

public:

bool move(int hangmoi, int cotmoi, QuanCo\* \*\*banco)

{

// Khong di chuyen duoc neu quan co o dich den cung mau

if((banco[hangmoi][cotmoi] != NULL) && (banco[hangmoi][cotmoi] -> trang==trang))

return false;

if(((hangmoi - hang == 1) && (cotmoi - cot == 1)) ||

((hangmoi - hang == 1) && (cotmoi - cot == 0)) ||

((hangmoi - hang == 1) && (cotmoi - cot == -1)) ||

((hangmoi - hang == 0) && (cotmoi - cot == -1)) ||

((hangmoi - hang == 0) && (cotmoi - cot == 1)) ||

((hangmoi - hang == -1) && (cotmoi - cot == -1)) ||

((hangmoi - hang == -1) && (cotmoi - cot == 0)) ||

((hangmoi - hang == -1) && (cotmoi - cot == 1)))

{

return true;

}

else

return false;

}

};

class Hau: public QuanCo{

public:

bool move(int hangmoi, int cotmoi, QuanCo\* \*\*banco)

{

// Khong di chuyen duoc neu quan co o dich den cung mau

if((banco[hangmoi][cotmoi] != NULL) && (banco[hangmoi][cotmoi] -> trang == trang))

return false;

// Cac buoc di kha di cua Hau. No chinh la su ket hop cac kieu di cua Xe va Tuong.

if((hangmoi - hang == 0) && (cotmoi - cot > 0))

{

int soBuocDi = cotmoi - cot;

if(soBuocDi == 1)

return true;

if(soBuocDi > 1)

{

for(int i = 1; i <= (soBuocDi-1); i++)

{

if(banco[hang][cot + i] != NULL)

return false;

}

return true;

}

}

else if((hangmoi - hang == 0) && (cotmoi - cot < 0))

{

int soBuocDi = cot - cotmoi;

if(soBuocDi == 1)

return true;

if(soBuocDi > 1)

{

for(int i = 1; i <= (soBuocDi-1); i++)

{

if(banco[hang][cot - i] != NULL)

return false;

}

return true;

}

}

else if((hangmoi - hang > 0) && (cotmoi - cot == 0))

{

int soBuocDi = hangmoi - hang;

if(soBuocDi == 1)

return true;

if(soBuocDi > 1)

{

for(int i = 1; i <= (soBuocDi-1); i++)

{

if(banco[hang + i][cot] != NULL)

return false;

}

return true;

}

}

else if((hangmoi - hang < 0) && (cotmoi - cot == 0))

{

int soBuocDi = hang - hangmoi;

if(soBuocDi == 1)

return true;

if(soBuocDi > 1)

{

for(int i = 1; i <= (soBuocDi-1); i++)

{

if(banco[hang - i][cot] != NULL)

return false;

}

return true;

}

}

else if((hangmoi - hang > 0) && (cotmoi - cot > 0))

{

int soBuocHang = hangmoi - hang;

int soBuocCot = cotmoi - cot;

if(soBuocHang != soBuocCot)

return false;

if(soBuocHang == 1)

return true;

if(soBuocHang > 1)

{

for(int i = 1; i <= (soBuocHang-1); i++)

{

if(banco[hang + i][cot + i] != NULL)

return false;

}

return true;

}

}

else if((hangmoi - hang > 0) && (cotmoi - cot < 0))

{

int soBuocHang = hangmoi - hang;

int soBuocCot = cot - cotmoi;

if(soBuocHang != soBuocCot)

return false;

if(soBuocHang == 1)

return true;

if(soBuocHang > 1)

{

for(int i = 1; i <= (soBuocHang-1); i++)

{

if(banco[hang + i][cot - i] != NULL)

return false;

}

return true;

}

}

else if((hangmoi - hang < 0) && (cotmoi - cot > 0))

{

int soBuocHang = hang - hangmoi;

int soBuocCot = cotmoi - cot;

if(soBuocHang != soBuocCot)

return false;

if(soBuocHang == 1)

return true;

if(soBuocHang > 1)

{

for(int i = 1; i <= (soBuocHang-1); i++)

{

if(banco[hang - i][cot + i] != NULL)

return false;

}

return true;

}

}

else if((hangmoi - hang < 0) && (cotmoi - cot < 0))

{

int soBuocHang = hang - hangmoi;

int soBuocCot = cot - cotmoi;

if(soBuocHang != soBuocCot)

return false;

if(soBuocHang == 1)

return true;

if(soBuocHang > 1)

{

for(int i = 1; i <= (soBuocHang-1); i++)

{

if(banco[hang - i][cot - i] != NULL)

return false;

}

return true;

}

}

else

return false;

return false;

}

};

class Xe: public QuanCo{

public:

bool move(int hangmoi, int cotmoi, QuanCo\* \*\*banco)

{

// Khong di chuyen duoc neu quan co o dich den cung mau

if((banco[hangmoi][cotmoi] != NULL) && (banco[hangmoi][cotmoi] -> trang == trang))

return false;

// Cac buoc di hop le cua Xe.

// Mot trong hieu cua (hangmoi - hang) hoac (cotmoi - cot) phai = 0 vi Xe xhi di chuyen hoac ngang hoac doc cho moi buoc di.

if((hangmoi - hang == 0) && (cotmoi - cot > 0))// truong hop hang khong thay doi

{

int soBuocDi = cotmoi - cot;

if(soBuocDi == 1) // Xe di chuyen 1 cot hoac an con co o dich den.

return true;

// Neu xe di chuyen nhieu hon 1 cot, kiem tra xem tren duong di co con co nao khong?

if(soBuocDi > 1)

{

for(int i = 1; i <= (soBuocDi-1); i++)

{

if(banco[hang][cot + i] != NULL)

return false;

}

return true;

}

}

else if((hangmoi - hang == 0) && (cotmoi - cot < 0))

{

int soBuocDi = cot - cotmoi;

if(soBuocDi == 1)

return true;

if(soBuocDi > 1)

{

for(int i = 1; i <= (soBuocDi-1); i++)

{

if(banco[hang][cot - i] != NULL)

return false;

}

return 0;

}

}

else if((hangmoi - hang > 0) && (cotmoi - cot == 0))

{

int soBuocDi = hangmoi - hang;

if(soBuocDi == 1)

return true;

if(soBuocDi > 1)

{

for(int i = 1; i <= (soBuocDi-1); i++)

{

if(banco[hang + i][cot] != NULL)

return false;

}

return true;

}

}

else if((hangmoi - hang < 0) && (cotmoi - cot == 0))

{

int soBuocDi = hang - hangmoi;

if(soBuocDi == 1)

return true;

if(soBuocDi > 1)

{

for(int i = 1; i <= (soBuocDi-1); i++)

{

if(banco[hang - i][cot] != NULL)

return false;

}

return true;

}

}

else

return false;

return false;

}

};

class Tuong: public QuanCo{

public:

bool move(int hangmoi, int cotmoi, QuanCo\* \*\*banco)

{

// Khong di chuyen duoc neu quan co o dich den cung mau

if((banco[hangmoi][cotmoi] != NULL) && (banco[hangmoi][cotmoi] -> trang == trang))

return false;

// Kiem tra cac buoc di hop le cua Tuong. Co 4 cach Tuong co the di: Tren Phai, Duoi Phai, Duoi Trai, Tren Trai.

if((hangmoi - hang > 0) && (cotmoi - cot > 0))

{

int soBuocHang = hangmoi - hang;

int soBuocCot = cotmoi - cot;

// Tuong chi co the di cheo nen soBuocHang bang soBuocCot

if(soBuocHang != soBuocCot)

return false;

if(soBuocHang == 1) // Neu chi di mot buoc thi co the an con co o dich den hoac di chuyen len tren mot buoc.

return true;

// Neu di chuyen nhieu hon 1 buoc, kiem tra xem tren duong di co con co nao khong?

if(soBuocHang > 1)

{

for(int i = 1; i <= (soBuocHang-1); i++)

{

if(banco[hang + i][cot + i] != NULL)

return false;

}

return true;

}

}

else if((hangmoi - hang > 0) && (cotmoi - cot < 0))

{

int soBuocHang = hangmoi - hang;

int soBuocCot = cot - cotmoi;

if(soBuocHang != soBuocCot)

return false;

if(soBuocHang == 1)

return true;

if(soBuocHang > 1)

{

for(int i = 1; i <= (soBuocHang-1); i++)

{

if(banco[hang + i][cot - i] != NULL)

return false;

}

return true;

}

}

else if((hangmoi - hang < 0) && (cotmoi - cot > 0))

{

int soBuocHang = hang - hangmoi;

int soBuocCot = cotmoi - cot;

if(soBuocHang != soBuocCot)

return false;

if(soBuocHang == 1)

return true;

if(soBuocHang > 1)

{

for(int i = 1; i <= (soBuocHang-1); i++)

{

if(banco[hang - i][cot + i] != NULL)

return false;

}

return true;

}

}

else if((hangmoi - hang < 0) && (cotmoi - cot < 0))

{

int soBuocHang = hang - hangmoi;

int soBuocCot = cot - cotmoi;

if(soBuocHang != soBuocCot)

return false;

if(soBuocHang == 1)

return true;

if(soBuocHang > 1)

{

for(int i = 1; i <= (soBuocHang-1); i++)

{

if(banco[hang - i][cot - i] != NULL)

return false;

}

return true;

}

}

else

return false;

return false;

}

};

class Ma: public QuanCo{

public:

bool move(int hangmoi, int cotmoi, QuanCo\* \*\*banco)

{

// Khong di chuyen duoc neu quan co o dich den cung mau

if((banco[hangmoi][cotmoi] != NULL) && (banco[hangmoi][cotmoi] -> trang == trang))

return false;

if(((hangmoi - hang == 2) && (cotmoi - cot == 1)) ||

((hangmoi - hang == 2) && (cotmoi - cot == -1)) ||

((hangmoi - hang == 1) && (cotmoi - cot == 2)) ||

((hangmoi - hang == 1) && (cotmoi - cot == -2)) ||

((hangmoi - hang == -1) && (cotmoi - cot == 2)) ||

((hangmoi - hang == -1) && (cotmoi - cot == -2)) ||

((hangmoi - hang == -2) && (cotmoi - cot == 1)) ||

((hangmoi - hang == -2) && (cotmoi - cot == -1)))

{

return true;

}

else

return false;

}

};

class Chot: public QuanCo{

public:

bool move(int hangmoi, int cotmoi, QuanCo\* \*\*banco)

{

//Neu la chot trang thi hang xuat phat la 1, va di chuyen thi hang se tang

int xuatphat = 1;

int inc = 1;

//Neu la chot den thi hang xuat phat la 6, va di chuyen thi hang se giam

if(!trang)

{

xuatphat = 6;

inc = -1;

}

if(((hang == xuatphat) && ((hangmoi - hang) == (inc\*2)) && (banco[hang + (inc\*1)][cot] == NULL) && (cotmoi - cot == 0) && (banco[hangmoi][cotmoi] == NULL))

|| ((hangmoi - hang == (inc\*1)) && (banco[hangmoi][cotmoi] == NULL) && (cotmoi - cot == 0))

|| (((cotmoi - cot == 1) || (cotmoi - cot == -1)) && (hangmoi - hang == (inc\*1)) && (banco[hangmoi][cotmoi] != NULL) && (banco[hangmoi][cotmoi] -> trang != trang)))

{

return true;

}

else

return false;

}

};

class BanCo{

private:

QuanCo\* \*\*banco;

Vua \*vua;

Hau \*hau;

Xe \*xe;

Tuong \*tuong;

Ma \*ma;

Chot \*chot;

int buocdi;

public:

BanCo();

~BanCo();

void init();

void display();

void LuuBanCo(string);

void NapBanCo(string);

void GanQuanCo(string,int,int);

bool move(string tu, string den);

int SoBuocDi();

bool Checked(bool trang);

vector<int> KingPos(bool trang);

};

BanCo::BanCo()

{

init();

}

int BanCo::SoBuocDi()

{

return buocdi;

}

bool BanCo::Checked(bool trang)

{

vector<int> ViTriCuaVua = KingPos(trang);

int den\_hang =ViTriCuaVua[0];

int den\_cot = ViTriCuaVua[1];

for(int i = 0; i < 8; i++)

for(int j = 0; j < 8; j++)

if(banco[i][j] != NULL)

if(banco[i][j]->trang!=trang)

if(banco[i][j] -> move(den\_hang, den\_cot, banco))

return true;

return false;

}

vector<int> BanCo::KingPos(bool trang)

{

vector<int>ret;

for(int i = 0; i < 8; i++)

for(int j = 0; j < 8; j++)

if(banco[i][j] != NULL)

if(banco[i][j]->ten.compare("Vua")==0)

if(banco[i][j]->trang==trang)

{

ret.push\_back(i);

ret.push\_back(j);

break;

}

return ret ;

}

bool BanCo::move(string tu, string den)

{

int tu\_hang = (int)(tu[1] - '1');

int tu\_cot = (int)(tu[0] - 'A');

int den\_hang = (int)(den[1] - '1');

int den\_cot = (int)(den[0] - 'A');

string Mau = (banco[tu\_hang][tu\_cot]->trang)?"trang":"den";

string Ten = banco[tu\_hang][tu\_cot]->ten;

if((((buocdi%2)==0) && (Mau=="den"))||(((buocdi%2)==1) && (Mau=="trang")))

{

cout << "Khong the di chuyen " << Ten << " " << Mau << " tu " << tu << " den " << den << " vi khong phai luot di cua ben " << Mau << endl << endl;

return false;

}

else if( banco[tu\_hang][tu\_cot] -> move(den\_hang, den\_cot, banco))

{

QuanCo\* FROM = banco[tu\_hang][tu\_cot];

QuanCo\* TO = banco[den\_hang][den\_cot];

GanQuanCo(FROM->ten+ ((FROM->trang)? "t" : "d"),den\_hang,den\_cot);

banco[tu\_hang][tu\_cot] = NULL;

if(banco[den\_hang][den\_cot]->ten.compare("Chot")==0 && ((banco[den\_hang][den\_cot]->trang && den\_hang==7)||(!(banco[den\_hang][den\_cot]->trang) && den\_hang==0)))

{

bool on=true;

string cell="";

int chon;

while(on)

{

cout << "Ban muon Chot thanh quan nao? 0: Chot, 1: Hau, 2: Xe, 3: Tuong, 4: Ma"<<endl;

cin >> chon;

switch(chon)

{

case 0: cell="Chot";on = false;break;

case 1: cell="Hau";on = false;break;

case 2: cell="Xe";on = false;break;

case 3: cell="Tuong";on = false;break;

case 4: cell="Ma";on = false;break;

default: cout << endl <<"Nhap sai, xin moi nhap lai."<<endl;break;

}

}

if(chon>0)

{

string mau = "d";

if(banco[den\_hang][den\_cot]->trang)

mau="t";

GanQuanCo(cell + mau,den\_hang,den\_cot);

cout << "Chot " << Mau << " da duoc phong " << cell << " tai " << den << "." << endl << endl;

}

}

if(Checked(FROM->trang))

{

cout << "Di chuyen khong hop le vi quan Vua bi chieu." << endl << endl;

return false;

}

else

{

//Buoc di hop le, tang 1 buoc

buocdi++;

string Ten = FROM->ten;

cout << "Buoc di " << buocdi << ". Ben " << Mau << " di chuyen: " << Ten << " tu " << tu << " den " << den << endl;

if(TO != NULL)

cout << "Quan co " << TO->ten << " " << ((TO->trang)? "trang" : "den") << " da bi loai boi " << Ten << " " << Mau << endl;

if(Checked(!(FROM->trang)))

cout << "BEN " << ((FROM->trang)? "DEN" : "TRANG") << " DANG BI CHECKMATE" << endl;

}

return true;

}

else

{

cout << "Khong the di chuyen " << Ten << " " << Mau << " tu " << tu << " den " << den << " vi nuoc di khong hop le." << endl << endl;

return false;

}

}

BanCo::~BanCo()

{

delete [] vua;

delete [] hau;

delete [] xe;

delete [] tuong;

delete [] ma;

delete [] chot;

for(int i = 0; i < 8; i++)

delete [] banco[i];

delete [] banco;

}

void BanCo::init()

{

buocdi = 0;

banco = new QuanCo\*\*[8];

for(int i = 0; i < 8; i++)

banco[i] = new QuanCo\*[8];

for(int i = 0; i < 8; i++)

for(int j = 0; j < 8; j++)

banco[i][j] = NULL;

vua = new Vua[2];

hau = new Hau[2];

xe = new Xe[4];

tuong = new Tuong[4];

ma = new Ma[4];

chot = new Chot[16];

//vua trang

vua[0].ten = "Vua";

vua[0].trang = true;

vua[0].hang = 0;

vua[0].cot = 4;

banco[vua[0].hang][vua[0].cot] = &vua[0];

//hau trang

hau[0].ten = "Hau";

hau[0].trang = true;

hau[0].hang = 0;

hau[0].cot = 3;

banco[hau[0].hang][hau[0].cot] = &hau[0];

//vua den

vua[1].ten = "Vua";

vua[1].trang = false;

vua[1].hang = 7;

vua[1].cot = 4;

banco[vua[1].hang][vua[1].cot] = &vua[1];

//hau den

hau[1].ten = "Hau";

hau[1].trang = false;

hau[1].hang = 7;

hau[1].cot = 3;

banco[hau[1].hang][hau[1].cot] = &hau[1];

//xe tuong ma

for(int i = 0; i <= 3; i++)

{

xe[i].ten = "Xe";

tuong[i].ten = "Tuong";

ma[i].ten = "Ma";

//xe tuong ma quan trang

if(i < 2)

{

tuong[i].trang = true;

tuong[i].hang = 0;

ma[i].trang = true;

ma[i].hang = 0;

xe[i].trang = true;

xe[i].hang = 0;

}

//xe tuong ma quan den

if(i >= 2)

{

xe[i].trang = false;

xe[i].hang = 7;

tuong[i].trang = false;

tuong[i].hang = 7;

ma[i].trang = false;

ma[i].hang = 7;

}

if(i == 0 || i == 2)

{

xe[i].cot = 0;

tuong[i].cot = 2;

ma[i].cot = 1;

}

if(i == 1 || i == 3)

{

xe[i].cot = 7;

tuong[i].cot = 5;

ma[i].cot = 6;

}

banco[xe[i].hang][xe[i].cot] = &xe[i];

banco[tuong[i].hang][tuong[i].cot] = &tuong[i];

banco[ma[i].hang][ma[i].cot] = &ma[i];

}

//chot

for(int i = 0; i <= 15; i++)

{

chot[i].ten = "Chot";

if(i < 8)

{

chot[i].trang = true;

chot[i].hang = 1;

chot[i].cot = i;

}

if(i >= 8)

{

chot[i].trang = false;;

chot[i].hang = 6;

chot[i].cot = (i-8);

}

banco[chot[i].hang][chot[i].cot] = &chot[i];

}

}

void BanCo::display()

{

for(int i = 8; i >= 0; i--)

{

if(i == 8)

cout << " A B C D E F G H" << endl;

else

{

for(int j = 0; j <=8; j++)

{

if(j == 0)

cout << (i+1) << " ";

else

{

if(banco[i][j-1] == NULL)

cout << "\* ";

else

{

string Ten = banco[i][j-1]->ten;

string Mau= "d";

if(banco[i][j-1]->trang)

Mau= "t";

cout << Ten[0]<<Mau<< " ";

}

}

}

cout << endl;

}

}

cout << endl;

}

inline bool file\_exists (const std::string& name) {

ifstream f(name.c\_str());

if (f.good()) {

f.close();

return true;

} else {

f.close();

return false;

}

}

void BanCo::LuuBanCo(string dout)

{

//File banco.txt phai nam trong thu muc chua file CoVua.cpp

string filename=string(\_\_FILE\_\_);

string directory;

const size\_t last\_slash\_idx = filename.rfind('\\');

if (std::string::npos != last\_slash\_idx)

directory = filename.substr(0, last\_slash\_idx);

string file = directory + "\\" + dout;

ofstream myfile;

myfile.open(file);

for(int i = 0; i <=7; i++)

{

for(int j = 0; j <=7; j++)

{

if(banco[i][j] == NULL)

myfile << "\*";

else

{

string Ten = banco[i][j]->ten;

string Mau= "d";

if(banco[i][j]->trang)

Mau= "t";

myfile << Ten << Mau;

}

if(j<7)

myfile<< " ";

}

myfile << "\n";

}

myfile << buocdi;

myfile.close();

cout << "Da luu ban co xuong file " << file << endl;

}

void BanCo::GanQuanCo(string cell,int i, int j)

{

if(cell.compare("\*")==0)

{

banco[i][j] = NULL;

}

else

{

string ten = cell.substr(0,cell.length() - 1);

string mau = cell.substr(cell.length() - 1);

if(ten.compare("Vua")==0)

{

Vua\* quanco = new Vua[1];

quanco[0].ten = "Vua";

quanco[0].trang=true;

quanco[0].hang=i;

quanco[0].cot=j;

if(mau.compare("d")==0)

quanco[0].trang=false;

banco[i][j] = &quanco[0];

}

else if(ten.compare("Hau")==0)

{

Hau\* quanco = new Hau[1];

quanco[0].ten = "Hau";

quanco[0].trang=true;

quanco[0].hang=i;

quanco[0].cot=j;

if(mau.compare("d")==0)

quanco[0].trang=false;

banco[i][j] = &quanco[0];

}

else if(ten.compare("Xe")==0)

{

Xe\* quanco = new Xe[1];

quanco[0].ten = "Xe";

quanco[0].trang=true;

quanco[0].hang=i;

quanco[0].cot=j;

if(mau.compare("d")==0)

quanco[0].trang=false;

banco[i][j] = &quanco[0];

}

else if(ten.compare("Tuong")==0)

{

Tuong\* quanco = new Tuong[1];

quanco[0].ten = "Tuong";

quanco[0].trang=true;

quanco[0].hang=i;

quanco[0].cot=j;

if(mau.compare("d")==0)

quanco[0].trang=false;

banco[i][j] = &quanco[0];

}

else if(ten.compare("Ma")==0)

{

Ma\* quanco = new Ma[1];

quanco[0].ten = "Ma";

quanco[0].trang=true;

quanco[0].hang=i;

quanco[0].cot=j;

if(mau.compare("d")==0)

quanco[0].trang=false;

banco[i][j] = &quanco[0];

}

else if(ten.compare("Chot")==0)

{

Chot\* quanco = new Chot[1];

quanco[0].ten = "Chot";

quanco[0].trang=true;

quanco[0].hang=i;

quanco[0].cot=j;

if(mau.compare("d")==0)

quanco[0].trang=false;

banco[i][j] = &quanco[0];

}

}

}

void BanCo::NapBanCo(string din)

{

init();

//File banco.txt phai nam trong thu muc chua file CoVua.cpp

string filename=string(\_\_FILE\_\_);

string directory;

const size\_t last\_slash\_idx = filename.rfind('\\');

if (std::string::npos != last\_slash\_idx)

directory = filename.substr(0, last\_slash\_idx);

string file = directory + "\\" + din;

if(!file\_exists(file))

cout << "file khong ton tai!";

std::ifstream data(file);

std::string line;

int i=0;

while(std::getline(data,line))

{

if(!line.empty())

{

std::stringstream lineStream(line);

std::string cell;

int j = 0;

while(std::getline(lineStream,cell,' '))

{

if(i<8)

{

GanQuanCo(cell,i,j);

j++;

}

else

buocdi=stoi(cell);

}

}

i++;

if(i>8)

break;

}

cout << "Da nap ban co tu file " << file << endl;

}

void ChoiVanMoi(BanCo &banco)

{

banco.init();

banco.display();

cout<<"Co the thu cac buoc di A7-A5;A2-A5;A2-A4;B7-B6;A4-A5;B6-A5"<<endl;

while(true)

{

string tu,den;

cout << "Tu (0 de thoat): "; cin >> tu;

if(tu.compare("0")==0)

break;

cout << "Den: "; cin >> den;

if(!(tu.length()==2 && den.length()==2))

cout << "Nhap sai, xin moi nhap lai."<<endl;

else

{

bool contains\_non\_AH\_tu = !std::regex\_match(tu.substr(0,1), std::regex("^[A-H]+$"));

bool contains\_non\_AH\_den = !std::regex\_match(den.substr(0,1), std::regex("^[A-H]+$"));

bool contains\_non\_18\_tu = !std::regex\_match(tu.substr(1), std::regex("^[1-8]+$"));

bool contains\_non\_18\_den = !std::regex\_match(den.substr(1), std::regex("^[1-8]+$"));

if(contains\_non\_AH\_tu||contains\_non\_AH\_den||contains\_non\_18\_tu||contains\_non\_18\_den)

cout << "Nhap sai, xin moi nhap lai."<<endl;

else

if(banco.move(tu,den))

banco.display();

}

}

}

void NapGame(BanCo &banco)

{

banco.NapBanCo("banco.txt");

banco.display();

cout << "Da di duoc " << banco.SoBuocDi() << " nuoc. Den luot di cua ben ";

if((banco.SoBuocDi()%2)==0)

cout << "trang. " << endl;

else

cout << "den. " << endl;

while(true)

{

string tu,den;

cout << "Tu (0 de thoat): "; cin >> tu;

if(tu.compare("0")==0)

break;

cout << "Den: "; cin >> den;

if(!(tu.length()==2 && den.length()==2))

cout << "Nhap sai, xin moi nhap lai."<<endl;

else

{

bool contains\_non\_AH\_tu = !std::regex\_match(tu.substr(0,1), std::regex("^[A-H]+$"));

bool contains\_non\_AH\_den = !std::regex\_match(den.substr(0,1), std::regex("^[A-H]+$"));

bool contains\_non\_18\_tu = !std::regex\_match(tu.substr(1), std::regex("^[1-8]+$"));

bool contains\_non\_18\_den = !std::regex\_match(den.substr(1), std::regex("^[1-8]+$"));

if(contains\_non\_AH\_tu||contains\_non\_AH\_den||contains\_non\_18\_tu||contains\_non\_18\_den)

cout << "Nhap sai, xin moi nhap lai."<<endl;

else

if(banco.move(tu,den))

banco.display();

}

}

}

void LuuGame(BanCo &banco)

{

banco.LuuBanCo("banco.txt");

}

void Demo(BanCo &banco)

{

banco.init();

cout<<"Ban co ban dau:"<<endl;

banco.display();

string str;

banco.move("C2","C4"); banco.display(); getline(cin, str);

banco.move("B7","B5"); banco.display(); getline(cin, str);

banco.move("C4","B5"); banco.display(); getline(cin, str);

banco.move("G7","G5"); banco.display(); getline(cin, str);

banco.move("E2","E3"); banco.display(); getline(cin, str);

banco.move("H7","H5"); banco.display(); getline(cin, str);

banco.move("F1","E2"); banco.display(); getline(cin, str);

banco.move("G8","F6"); banco.display(); getline(cin, str);

banco.move("B2","B3"); banco.display(); getline(cin, str);

banco.move("F8","G7"); banco.display(); getline(cin, str);

banco.move("C1","B2"); banco.display(); getline(cin, str);

banco.move("G5","G4"); banco.display(); getline(cin, str);

banco.move("H2","H3"); banco.display(); getline(cin, str);

banco.move("C8","B7"); banco.display(); getline(cin, str);

banco.move("H1","H2"); banco.display(); getline(cin, str);

banco.move("A7","A6"); banco.display(); getline(cin, str);

banco.move("B5","A6"); banco.display(); getline(cin, str);

banco.move("B8","A6"); banco.display(); getline(cin, str);

banco.move("B1","C3"); banco.display(); getline(cin, str);

banco.move("D8","B8"); banco.display(); getline(cin, str);

banco.move("H3","G4"); banco.display(); getline(cin, str);

banco.move("C7","C5"); banco.display(); getline(cin, str);

banco.move("G2","G3"); banco.display(); getline(cin, str);

banco.move("H5","H4"); banco.display(); getline(cin, str);

banco.move("F2","F4"); banco.display(); getline(cin, str);

banco.move("H4","G3"); banco.display(); getline(cin, str);

banco.move("H2","H8"); banco.display(); getline(cin, str);

banco.move("G7","H8"); banco.display(); getline(cin, str);

banco.move("G4","G5"); banco.display(); getline(cin, str);

banco.move("F6","E4"); banco.display(); getline(cin, str);

banco.move("D1","C2"); banco.display(); getline(cin, str);

banco.move("A6","B4"); banco.display(); getline(cin, str);

banco.move("C2","B1"); banco.display(); getline(cin, str);

banco.move("E4","F2"); banco.display(); getline(cin, str);

banco.move("B1","H7"); banco.display(); getline(cin, str);

banco.move("D7","D6"); banco.display(); getline(cin, str);

banco.move("H7","H8"); banco.display(); getline(cin, str);

banco.move("E8","D7"); banco.display(); getline(cin, str);

banco.move("H8","B8"); banco.display(); getline(cin, str);

banco.move("A8","B8"); banco.display(); getline(cin, str);

banco.move("A1","B1"); banco.display(); getline(cin, str);

banco.move("B8","H8"); banco.display(); getline(cin, str);

banco.move("E2","F3"); banco.display(); getline(cin, str);

banco.move("B4","C2"); banco.display(); getline(cin, str);

banco.move("E1","E2"); banco.display(); getline(cin, str);

banco.move("B7","A6"); banco.display(); getline(cin, str);

banco.move("D2","D3"); banco.display(); getline(cin, str);

banco.move("H8","H2"); banco.display(); getline(cin, str);

banco.move("F3","G4"); banco.display(); getline(cin, str);

banco.move("F2","G4"); banco.display(); getline(cin, str);

banco.move("E2","F3"); banco.display(); getline(cin, str);

banco.move("C2","E3"); banco.display(); getline(cin, str);

banco.move("B2","C1"); banco.display(); getline(cin, str);

banco.move("A6","B7"); banco.display(); getline(cin, str);

banco.move("F3","G3"); banco.display(); getline(cin, str);

banco.move("H2","G2"); banco.display(); getline(cin, str);

banco.move("G3","H3"); banco.display(); getline(cin, str);

banco.move("G4","F2"); banco.display(); getline(cin, str);

banco.move("H3","H4"); banco.display(); getline(cin, str);

banco.move("E3","F5"); banco.display(); getline(cin, str);

banco.move("H4","H5"); banco.display(); getline(cin, str);

banco.move("G2","H2"); banco.display(); getline(cin, str);

banco.move("G1","H3"); banco.display(); getline(cin, str);

banco.move("H2","H3"); banco.display(); getline(cin, str);

cout<<"Ban co sau " << banco.SoBuocDi() << " nuoc di." << endl;

banco.display();

}

void PromotionCheck(BanCo &banco)

{

banco.init();

cout<<"Ban co ban dau:"<<endl;

banco.display();

string str;

banco.move("A2","A4"); banco.display(); getline(cin, str);

banco.move("H7","H6"); banco.display(); getline(cin, str);

banco.move("A4","A5"); banco.display(); getline(cin, str);

banco.move("G7","G6"); banco.display(); getline(cin, str);

banco.move("A5","A6"); banco.display(); getline(cin, str);

banco.move("F7","F6"); banco.display(); getline(cin, str);

banco.move("A6","B7"); banco.display(); getline(cin, str);

banco.move("E7","E6"); banco.display(); getline(cin, str);

banco.move("B7","A8"); banco.display(); getline(cin, str);

cout<<"Ban co sau " << banco.SoBuocDi() << " nuoc di." << endl;

banco.display();

}

void CheckMateCheck(BanCo &banco)

{

banco.init();

cout<<"Ban co ban dau:"<<endl;

banco.display();

string str;

banco.move("D2","D4"); banco.display(); getline(cin, str);

banco.move("G7","G5"); banco.display(); getline(cin, str);

banco.move("E2","E4"); banco.display(); getline(cin, str);

banco.move("F7","F6"); banco.display(); getline(cin, str);

banco.move("D1","H5"); banco.display(); getline(cin, str);

banco.move("D7","D6"); banco.display(); getline(cin, str);

// banco.move("A6","B7"); banco.display(); getline(cin, str);

// banco.move("E7","E6"); banco.display(); getline(cin, str);

// banco.move("B7","A8"); banco.display(); getline(cin, str);

cout<<"Ban co sau " << banco.SoBuocDi() << " nuoc di." << endl;

banco.display();

}

int Menu()

{

int c;

cout << endl << " CHUONG TRINH CO VUA: " << endl;

cout << "--------------------------------" << endl;

cout << "|1: Choi van co moi. |" << endl;

cout << "|2: Nap game da choi truoc do. |" << endl;

cout << "|3: Luu game xuong tap tin. |" << endl;

cout << "|4: Chay demo (> 60 nuoc). |" << endl;

cout << "|5: Kiem tra phong quan nang. |" << endl;

cout << "|6: Kiem tra CheckMate. |" << endl;

cout << "|0: Thoat. |" << endl;

cout << "--------------------------------" << endl;

cout << "Xin moi ban chon: ";

cin >> c;

return c;

}

int main() {

//banco.display();

BanCo banco;

int chon;

bool t = true;

do

{

chon = Menu();

switch(chon)

{

case 0:

cout << "Ket thuc chuong trinh." << endl;

t = false;

break;

case 1:

{

ChoiVanMoi(banco);

}

break;

case 2:

{

NapGame(banco);

}

break;

case 3:

{

LuuGame(banco);

}

break;

case 4:

{

Demo(banco);

}

break;

case 5:

{

PromotionCheck(banco);

}

break;

case 6:

{

CheckMateCheck(banco);

}

break;

default:

cout << endl <<"Nhap sai, xin moi nhap lai."<<endl;

break;

}

} while(t);

return 0;

}