#include <iostream>

#include <string>

using *namespace* std;

*struct* HocSinh

{

*int* MaHS;

    string HoTen;

    string NgaySinh;

    HocSinh \*Next;

};

*struct* Lop

{

    string TenLop;

    HocSinh \*DSHS; *// Danh sách học sinh trong lớp*

    Lop \*Down; *// Con trỏ đến lớp tiếp theo*

};

Lop \*F = NULL;

Lop *\**addLop(string *tenLop*)

{

    Lop \*newLop = new Lop();

    newLop->TenLop = *tenLop*;

    newLop->DSHS = NULL;

    newLop->Down = NULL;

    return newLop;

}

*void* chenLopLast(Lop *\*&F*, string *tenLop*)

{

    Lop \*newLop = addLop(*tenLop*);

    if (*F* == NULL)

    {

*F* = newLop;

        return;

    }

    Lop \*cur = *F*;

    while (cur->Down != NULL)

    {

        cur = cur->Down;

    }

    cur->Down = newLop;

}

Lop *\**DiaChi(Lop *\*F*, string *Blop*)

{

    if (*F* == NULL)

    {

        return NULL;

    }

    Lop \*cur = *F*;

    while (cur != NULL)

    {

        if (cur->TenLop == *Blop*)

        {

            return cur;

        }

        cur = cur->Down;

    }

    return NULL;

}

*void* inLop(Lop *\*F*)

{

    Lop \*cur = *F*;

    while (cur != NULL)

    {

        cout << cur->TenLop << " ";

        cur = cur->Down;

    }

}

*// SiSo(Lop \*F, St8 Blop) trả về sĩ số của lớp Blop trong danh sách F. Nếu không tìm thấy lớp Blop thì trả về giá trị 0.*

*int* siSo(Lop *\*F*, string *BLop*)

{

    if (*F* == NULL)

    {

        return 0;

    }

    Lop \*cur = *F*;

*int* c = 0;

    while (cur != NULL)

    {

        if (cur->TenLop == *BLop*)

        {

            HocSinh \*hs = cur->DSHS;

            while (hs != NULL)

            {

                c++;

                hs = hs->Next;

            }

        }

        cur = cur->Down;

    }

    return c;

}

*// -------------------------------HOC SINH-------------------*

*void* addHocSinhVaoLop(Lop *\*&F*, string *tenlop*, *int* *mahs*, string *hoten*, string *ngaysinh*)

{

    Lop \*lop = DiaChi(*F*, *tenlop*);

    if (lop != NULL)

    {

        HocSinh \*newHs = new HocSinh();

        newHs->MaHS = *mahs*;

        newHs->HoTen = *hoten*;

        newHs->NgaySinh = *ngaysinh*;

        newHs->Next = NULL;

        if (lop->DSHS == NULL)

        {

            lop->DSHS = newHs;

        }

        else

        {

            HocSinh \*curHs = lop->DSHS;

            while (curHs->Next != NULL)

            {

                curHs = curHs->Next;

            }

            curHs->Next = newHs;

        }

    }

}

*void* themhsLast(*int* *mhs*, string *hoten*, string *ngaysinh*)

{

}

*void* inHsTheoLop(Lop *\*F*, string *tenLop*)

{

    Lop \*cur = *F*;

    while (cur != NULL)

    {

        if (cur->TenLop == *tenLop*)

        {

            HocSinh \*curHs = cur->DSHS;

            while (curHs != NULL)

            {

                cout << curHs->HoTen << endl;

                curHs = curHs->Next;

            }

        }

        cur = cur->Down;

    }

}

*// ) BoSung(Lop \*&F, string Blop, int Bmahs, string Bhoten, string Bngaysinh) cho phép bổ sung một nút vào cuối danh sách học sinh của lớp Blop, sao cho nút đó có mã học sinh là Bmahs, họ tên là Bhoten và ngày sinh là Bngaysinh. Cụ thể ta xét các trường hợp:*

*// • Blop chưa có trong danh sách lớp thì chèn vào một lớp mới gồm có 1 học sinh với thông tin đã cho.*

*// • Blop đã có trong danh sách lớp, nhưng:*

*// o Bmahs chưa có trong danh sách học sinh thì chèn vào cuối danh sách 1 học sinh với thông tin đã cho.*

*// o Bmahs đã có trong danh sách học sinh thì cập nhật lại thông tin của nút này theo thông tin mới đã cho (cập nhật lại trường HoTen và NgaySinh).*

*void* boSung(Lop *\*&F*, string *Blop*, *int* *mhs*, string *hoten*, string *ns*)

{

    Lop \*lop = DiaChi(*F*, *Blop*);

    if (lop == NULL)

    {

        chenLopLast(*F*, *Blop*);

        addHocSinhVaoLop(*F*, *Blop*, *mhs*, *hoten*, *ns*);

        return;

    }

*// Lớp đã tồn tại => duyệt danh sách học sinh của lớp đó*

    HocSinh \*curHs = lop->DSHS;

    while (curHs != NULL)

    {

        if (curHs->MaHS == *mhs*)

        {

*// Nếu đã tồn tại học sinh => cập nhật thông tin*

            curHs->HoTen = *hoten*;

            curHs->NgaySinh = *ns*;

            return;

        }

        curHs = curHs->Next;

    }

    addHocSinhVaoLop(*F*, *Blop*, *mhs*, *hoten*, *ns*);

    return;

}

*// g) Xét một danh sách học sinh L (nút đầu trỏ bởi L), viết hàm:*

*//  XoaHS(HocSinh \* &L, int Bmahs) nhằm xoá một nút thuộc danh sách học sinh L có mã học sinh là Bmahs.*

*void* xoaHs(HocSinh *\*&F*, *int* *Bmhs*)

{

    if (*F* == NULL)

    {

        return;

    }

    HocSinh \*curHs = *F*;

    HocSinh \*next = NULL;

    HocSinh \*pre = NULL;

    while (curHs != NULL)

    {

        next = curHs->Next;

        if (curHs->MaHS == *Bmhs*)

        {

*F* = next;

            delete curHs;

            curHs = *F*;

        }

        else

        {

            pre->Next = next;

            delete curHs;

            curHs = next;

        }

        pre = curHs;

        curHs = next;

    }

}

*void* dsHs()

{

    Lop \*lop = F;

    HocSinh \*hs = lop->DSHS;

    while (hs != NULL)

    {

        cout << hs->HoTen;

        hs = hs->Next;

    }

}

*int* main()

{

    chenLopLast(F, "12A4");

    chenLopLast(F, "12B6");

    addHocSinhVaoLop(F, "12B6", 1, "huuThanh", "12/12/2005");

    addHocSinhVaoLop(F, "12B6", 2, "Doanh", "12/9/2005");

    addHocSinhVaoLop(F, "12B6", 3, "NgocSon", "1/12/2005");

    addHocSinhVaoLop(F, "12A4", 4, "Phuc", "1/12/2005");

    addHocSinhVaoLop(F, "12A4", 9, "Bao", "1/12/2005");

    addHocSinhVaoLop(F, "12A4", 8, "Khanh", "1/12/2005");

*// inLop(F);*

*// cout << endl;*

*// inHsTheoLop(F, "12A4");*

*// // cout << endl;*

*// bool c = checkExistClass(F, "12B1");*

*// cout << c;*

*// boSung(F, "12A4", 4, "phucUpdate", "11/11/1111");*

*// inHsTheoLop(F, "12A4");*

*// cout << "Si so cua lop 12B6 " << siSo(F, "12A4");*

    dsHs();

}