#include <iostream>

#include <string.h>

using namespace std;

struct node

{

int num = 1;

int fill = 0;

char name[3][10];

node \*next;

node \*prev;

};

class hostel

{

node \*header[3];

node \*cn;

public:

hostel()

{

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

header[i] = NULL;

}

void create()

{

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

{

for (int j = 0; j < 9; j++)

{

node \*nn = new node;

nn->next = NULL;

nn->prev = NULL;

if (header[i] == NULL)

{

header[i] = nn;

nn->num = 1;

}

else

{

node \*cn = header[i];

while (cn->next != NULL)

{

cn = cn->next;

}

if (j == 3 || j == 5 || j == 7 || j == 8)

{

cn->num = 3;

}

if (j == 2 || j == 4 || j == 6)

{

cn->num = 2;

}

cn->next = nn;

nn->prev = cn;

}

}

}

}

void display()

{

int j = 0, k = 0, l = 0;

for (int i = 0; i < 48; i++)

{

cout << "--";

}

cout << "\n "

;

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

{

cout << "| Floor number : "<< i << " \t\t";

}

cout << "|\n" ;

for (int i = 0; i < 48; i++)

{

cout << "--";

}

cn = header[j];

node \*sn = header[j + 1];

node \*tn = header[j + 2];

cout << "\n ";

while (cn != NULL)

{

if (cn->fill != cn->num && cn->num != 0)

{

j++;

cout << "| room no : "<< j;

cout << "->Vacant cots->"<< cn->num;

}

else

{

j++;

cout << "| room no : "<< j;

cout << "->Present ";

}

if (sn->fill != sn->num && sn->num != 0)

{

k++;

cout << "\t | room no : "<< j;

cout << "->Vacant cots->"<< sn->num;

}

else

{

k++;

cout << " \t | room no : "<< j;

cout << "->Present ";

}

if (tn->fill != tn->num && tn->num != 0)

{

l++;

cout << "\t | room no : "<< j;

cout << "->Vacant cots->"<< tn->num << "| ";

}

else

{

l++;

cout << "\t | room no : "<< j;

cout << "->Present "<< " | ";

}

cout << " \n ";

for (int i = 0; i < 48; i++)

{

cout << "--" ;

}

cout << "\n ";

cn = cn->next;

sn = sn->next;

tn = tn->next;

}

}

void book(int people)

{

int floor, room;

cout << "\nEnter the floor number : ";

cin >> floor;

try

{

if (floor < 0 || floor > 4)

{

throw(floor);

}

cn = header[floor - 1];

cout << "\nEnter the room number : ";

cin >> room;

try

{

if (room < 0 || room > 10)

{

throw(room);

}

else

{

int i = 1;

while (i < room)

{

cn = cn->next;

i++;

}

if (cn->num >= people)

{

cout << "\nroom is vacant you can apply for room" ;

int count = 0;

while (cn->fill - 1 <= cn->num)

{

cout << "\nEnter name "<< cn -> fill + 1 << " : ";

cin >> cn->name[cn->fill];

count++;

cn->fill++;

if (count >= people)

{

break;

}

}

cn->num = cn->num - people;

}

else

{

cout << "\nroom is not vacant... SORRY !!!";

}

}

}

catch (int r)

{

cout << "\ninvalid room number : "<< r;

}

}

catch (int r)

{

cout << " \ninvalid floor number : " << r;

}

}

void cancle(int check)

{

char namecheck[10];

int flag = 0;

int room, i = 1;

try

{

if (check < 0 || check > 4)

{

throw(check);

}

else

{

cout << " Enter the room no : ";

cin >> room;

try

{

if (room < 0 || room > 10)

{

throw(room);

}

else

{

cout << " Enter the name to be delete :";

cin >> namecheck;

cn = header[check - 1];

while (i < room)

{

cn = cn->next;

i++;

}

i = 0;

while (i < 3)

{

if (!strcmp(namecheck, cn -> name[i]))

{

flag = 1;

break;

i = 0;

}

else

i++;

}

if (flag == 1 && cn->fill != 0)

{

cout << "\nrecord deleted : "<< cn -> name[i];

cn->name[i][0] ='A';

cn->name[i][1] ='\0';

cn->fill--;

cn->num++;

}

else

cout << "\nrecord not present ";

}

}

catch (int r)

{

cout << "\ninvalid room number : " << r;

}

}

}

catch (int r)

{

cout << " \n floor dosn't exist : " << r;

}

}

void upgrade(int check)

{

char namecheck[10];

int room, i = 1;

try

{

if (check < 0 || check > 4)

{

throw(check);

}

else

{

cout << " Enter the room no : ";

cin >> room;

try

{

if (room < 0 || room > 10)

{

throw(room);

}

else

{

cout << "Enter the name to be updated :";

cin >> namecheck;

cn = header[check - 1];

while (i < room)

{

cn = cn->next;

i++;

}

i = 0;

while (i < 3)

{

if (!strcmp(namecheck, cn -> name[i]))

{

cout << "\nenter updated name : " ;

cin >> cn->name[i];

break;

}

else

i++;

}

if (i >= 3)

cout << "record not found ";

else

{

cout << "\nrecord updated\nprevious name : "<< namecheck << "\nupdated name : "<< cn->name[i];

}

}

}

catch (int r)

{

cout << "\ninvalid room number : "<< r;

}

}

}

catch (int r)

{

cout << "\n floor dosn't exist : "<< r;

}

}

};

int main()

{

hostel obj;

int key;

char ch;

int floorcheck;

obj.create();

do

{

cout << "\n1.Book a room for 1 person\n2.Book a room for 2person\n3.Book a room for 3 person\n4.Display the current status of rooms\n5.cancle a cot\n6.upgrade"<< endl;

cout << " Enter your choice : " ;

cin >> key;

switch (key)

{

case 1:

{

obj.book(1);

break;

}

case 2:

{

obj.book(2);

break;

}

case 3:

{

obj.book(3);

break;

}

case 4:

{

obj.display();

break;

}

case 5:

{

cout << "Enter floor number : ";

cin >> floorcheck;

obj.cancle(floorcheck);

break;

}

case 6:

{

cout << "Enter floor number : ";

cin >> floorcheck;

obj.upgrade(floorcheck);

break;

}

default:

cout << "\nInvalid choice ";

}

cout << "\nDo you want to continue(Y / N) ";

cin >> ch;

} while (ch =='Y'|| ch =='y');

}