#include <iostream>

#include <string.h>

#include <math.h>

using namespace std;

class FlightBooking {

public:

FlightBooking(int id, int capacity, int reserved);

void printStatus();

void reserveSeats(int ticket);

void cancelReservations(int);

bool Checker();

int prectangle(int, int);

private:

int id;

int capacity;

int reserved;

int reserve;

int ticket;

void set(int, int, int);

void get();

};

FlightBooking::FlightBooking(int id, int capacity, int reserved) {

set(id, capacity, reserved);

}

void FlightBooking::set(int id, int capacity, int reserved) {

this->id = id;

this->capacity = capacity;

this->reserved = reserved;

}

int FlightBooking::prectangle(int capacity, int reserved) {

return (reserved \* 100) / capacity;

}

void FlightBooking::printStatus() {

cout << "Flight" << this->id << ":" << this->reserved << "/" << this->capacity

<< " "

<< "(" << prectangle(capacity, reserved) << "%) seats reserved" << endl;

}

void FlightBooking::reserveSeats(int ticket) {

this -> reserve = this -> reserved + ticket;

if(!Checker()){

if(106 > prectangle(this ->capacity, this -> reserve)){

this->reserved += ticket;

goto Success;

}

}

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

cout << "---Cannot perform this operation---" << endl;

Success:

ticket = ticket;

}

void FlightBooking::cancelReservations(int Number) {

if(0 <= this -> reserved - Number){

this->reserved -= Number;

goto Success;

}

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

cout << "---Cannot perform this operation---" << endl;

Success:

Number = Number;

}

bool FlightBooking::Checker(){

if(prectangle(this -> capacity, this -> reserved)>106){

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

cout << "---Cannot perform this operation---" << endl;

return true;

}

if(this -> capacity < 0 || this -> reserved < 0){

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

cout << "---Cannot perform this operation---" << endl;

return true;

}

return false;

}

int main(){

int reserved;

int capacity;

int argument = 0;

int number;

int j;

string term;

string command = "";

TryAgain:

cout << "Provide flight capacity: ";

cin >> capacity;

cout << "Provide number of reserved seats: ";

cin >> reserved;

FlightBooking booking(1, capacity, reserved);

if(booking.Checker()){

goto TryAgain;

}

while(command != "quit"){

argument = 0;

booking.printStatus();

cout << "What would you like to do?: ";

getline(cin, command);

if(command[0] == 'a'){

j = 0;

for(int i = command.length()-1; i >= 4; i--){

term = command[i];

number = stoi(term);

argument += number \* pow(10, j);

j++;

}

booking.reserveSeats(argument);

}else if (command[0] == 'c'){

j = 0;

for(int i = command.length()-1; i >= 7; i--){

term = command[i];

number = stoi(term);

argument += number \* pow(10, j);

j++;

}

booking.cancelReservations(argument);

}

}

}