

BÀI 9: QUẢN LÝ PHÒNG KHÁCH SẠN

Một khách sạn phân cấp các phòng theo nhiều loại và dựa trên thời gian thuê của từng khách để lập hóa đơn tiền phòng. Khai báo lớp Khách hàng (mã khách hàng, Họ tên, Địa chỉ, Số ĐT loại phòng cần thuê, số phòng cần thuê), mã khách hàng là một số nguyên có 3 chữ số, tự động tăng. Khai báo lớp Phòng gồm các thuộc tính (mã kiểu phòng, Kiểu phòng, Mức tiền thuê, Số phòng) - với kiểu phòng có thể là: phòng đơn, phòng đôi và phòng VIP, mã phòng là một số nguyên có 3 chữ số, tự động tăng. Khai báo lớp Bảng sắp xếp có quan hệ kết hợp (association) với lớp Khách hàng và lớp Phòng trong đó một khách hàng được sắp xếp tại loại phòng phù hợp (nếu thiếu thì đề nghị loại phòng khác) cùng với số lượng phòng loại đó và số ngày thuê. Viết chương trình trong ngôn ngữ C++ thực hiện các yêu cầu sau:

1. Nhập danh sách Phòng. In ra danh sách phòng đã có
2. Nhập danh sách Khách hàng. In ra danh sách các KH đã có
3. Nhập danh sách sắp xếp phòng cho mỗi khách hàng đã có trong và in danh sách ra màn hình. Chú ý kiểm tra số phòng còn lại mỗi kiểu.
4. Sắp xếp danh sách sắp xếp phòng :
 - a. Theo loại phòng
 - b. Theo số phòng cần thuê
5. Tính toán và lập hóa đơn cho mỗi khách hàng.

SOURCE CODE: [Github bài 9.](#)

```
1 #include <bits/stdc++.h>
2 using namespace std;
3
4 //Tao class KhachHang
5 class KhachHang{
6     private:
7         string HoTen,DiaChi,Sdt,loaiPhong;
8         int soPhong,maKH;
9         static int maKHt;
10    public:
11        KhachHang();
12        void NhapKH();
13        void InKH() const;
14        int getmaKH();
15        string getDiaChi();
16        string getHoTen();
17        int getsoPhong();
18        void giamsoPhong(int soluong);
19 };
20
21 //Khoi tao ma khach hang
22 int KhachHang::maKHt = 100;
23 KhachHang::KhachHang(){
24     HoTen = DiaChi = Sdt = loaiPhong = "";
25     soPhong = 0;
26     maKH = maKHt++;
27 }
28
29 //Ham nhap thong tin khach hang
30 void KhachHang::NhapKH(){
31     cout<<"Ho va ten: ", getline(cin,HoTen);
32     cout<<"Dia chi: ", getline(cin,DiaChi);
33     cout<<"So dien thoai: ", getline(cin,Sdt);
34     cout<<"So phong can thue: ", cin>>soPhong, cin.ignore();
35     cout<<"Loai phong: ", getline(cin,loaiPhong);
36 };
37
38 //Ham in thong tin khach hang
39 void KhachHang::InKH() const{
40     cout<<setw(20)<<left<<maKH<<setw(20)<<left<<HoTen;
41     cout<<setw(20)<<left<<DiaChi<<setw(20)<<left<<Sdt;
42     cout<<setw(20)<<left<<soPhong<<setw(20)<<left<<loaiPhong<<endl;
43 };
44
45 //Ham giam so luong phong can dat cua khach hang
46 void KhachHang::giamsoPhong(int soluong){
47     if(soPhong-soluong>=0) soPhong-=soluong;
48 }
49
50 //Lay ma khach hang, dia chi, ho ten va so phong can dat
```

```

51 int KhachHang::getmaKH() {return maKH;};
52 string KhachHang::getDiaChi() {return DiaChi;};
53 string KhachHang::getHoTen() {return HoTen;};
54 int KhachHang::getsoPhong() {return soPhong;};
55
56
57 //Tao class Phong
58 class Phong{
59     private:
60         string kieuPhong;
61         int muctienThue, soPhong, makieuPhong;
62         static int makieuPhongt;
63     public:
64         Phong();
65         void NhapPhong();
66         void InPhong() const;
67         void giamsoPhong(int soluong);
68         int getmakieuPhong();
69         string getkieuPhong();
70         int getmuctienThue();
71         int getsoPhong();
72 };
73
74 //Ham khoi tao ma kieu phong
75 int Phong::makieuPhongt = 100;
76 Phong::Phong() {
77     kieuPhong = "";
78     soPhong = muctienThue = 0;
79     makieuPhong = makieuPhongt++;
80 }
81
82 //Ham nhap thong tin cua phong cho thue
83 void Phong::NhapPhong() {
84     cout<<"Kieu phong(don, doi, vip): ", getline(cin, kieuPhong);
85     cout<<"So phong co the cho thue: ", cin>>soPhong;
86     cout<<"Tien thue phong: ", cin>>muctienThue, cin.ignore();
87     cout<<endl;
88 };
89
90 //Ham in ra thong tin phong
91 void Phong::InPhong() const{
92     cout<<setw(30)<<left<<makieuPhong<<setw(30)<<left<<kieuPhong;
93     cout<<setw(30)<<left<<soPhong<<setw(30)<<left<<muctienThue<<endl;
94 };
95
96 //Ham giam so luong phong khi khach hang dat
97 void Phong::giamsoPhong(int soluong) {
98     if(soPhong-soluong>=0) soPhong-=soluong;
99 }
100
101 //Lay ma kieu phong, kieu phong(don, doi, vip), so phong, gia thue moi phong
102 int Phong::getmakieuPhong() {return makieuPhong;};

```

```

103 string Phong::getkieuPhong(){return kieuPhong;}
104 int Phong::getsoPhong(){return soPhong;}
105 int Phong::getmuctienThue(){return muctienThue;}
106
107 //Tao class BangSapXep
108 class BangSapXep{
109     private:
110         KhachHang khachhang;
111         Phong phong;
112         int songayThue,soluongThue;
113     public:
114         void NhapP(KhachHang a, Phong b, int songayThue_, int soluongThue_);
115         void InBang();
116         int getsongayThue();
117         int getsoluongThue();
118         KhachHang getDataKhach();
119         Phong getDataPhong();
120         void XuatHoaDon();
121 };
122
123 //Ham nhap thong tin khach hang thue phong
124 void BangSapXep::NhapP(KhachHang a, Phong b, int songayThue_, int soluongThue_){
125     khachhang = a;
126     phong = b;
127     songayThue = songayThue_;
128     soluongThue = soluongThue_;
129 }
130
131 //In danh sach khach hang thue phong
132 void BangSapXep::InBang() {
133     cout<<setw(20)<<left<<phong.getmakieuPhong()<<setw(20)<<left<<khachhang.getHoTen();
134     cout<<setw(20)<<left<<phong.getkieuPhong()<<setw(20)<<left<<songayThue;
135     cout<<setw(20)<<left<<soluongThue<<setw(20)<<left<<phong.getmuctienThue()<<endl;
136 }
137
138 //Lay so ngay thue, so luong thue
139 int BangSapXep::getsongayThue(){return songayThue;};
140 int BangSapXep::getsoluongThue(){return soluongThue;};
141
142 //Lay du lieu cua phong va khach hang
143 Phong BangSapXep::getDataPhong(){return phong;};
144 KhachHang BangSapXep::getDataKhach(){return khachhang;};
145
146 //In hoa don cho moi khach hang
147 void BangSapXep::XuatHoaDon() {
148     cout<<setw(17)<<left<<khachhang.getmaKH()<<setw(17)<<left<<khachhang.getHoTen();
149     cout<<setw(17)<<left<<phong.getkieuPhong()<<setw(17)<<left<<songayThue<<setw(17)<<soluongThue;
150     cout<<setw(17)<<left<<phong.getmuctienThue()<<setw(17)<<left<<soluongThue*phong.getmuctienThue();
151 }
152
153
154 //Cac ham ve bang

```

```

155 static void InKhachHang() {
156     cout<<setfill('-');
157     cout<<setw(120)<<'-'<<endl;
158     cout<<setfill(' ');
159     cout<<setw(65)<<right<<"DANH SACH KHACH HANG"<<endl;
160     cout<<setfill('-');
161     cout<<setw(120)<<'-'<<endl;
162     cout<<setfill(' ');
163     cout<<setw(20)<<left<<"Ma Khach Hang"<<setw(20)<<left<<"Ho Va Ten";
164     cout<<setw(20)<< left<<"Dia Chi"<<setw(20)<<left<<"So Dien Thoai";
165     cout<<setw(20)<<left<<"So Phong Can Thue"<<setw(20)<<left<<"Loai Phong";
166     cout<<endl;
167     cout<<setfill('-');
168     cout<<setw(120)<<'-'<<endl;
169     cout<<setfill(' ');
170 }
171
172 static void InPhong() {
173     cout<<setfill('-');
174     cout<<setw(120)<<'-'<<endl;
175     cout<<setfill(' ');
176     cout<<setw(65)<<right<<"DANH SACH PHONG"<<endl;
177     cout<<setfill('-');
178     cout<<setw(120)<<'-'<<endl;
179     cout<<setfill(' ');
180     cout<<setw(30)<<left<<"Ma Kieu Phong"<<setw(30)<<left<<"Kieu Phong";
181     cout<<setw(30)<< left<<"So Phong Co The Cho Thue"<<setw(30)<<left<<"Gia Thue Phong";
182     cout<<endl;
183     cout<<setfill('-');
184     cout<<setw(120)<<'-'<<endl;
185     cout<<setfill(' ');
186 }
187
188 static void InPhongDaDat() {
189     cout<<setfill('-');
190     cout<<setw(120)<<'-'<<endl;
191     cout<<setfill(' ');
192     cout<<setw(65)<<right<<"DANH SACH DAT PHONG"<<endl;
193     cout<<setfill('-');
194     cout<<setw(120)<<'-'<<endl;
195     cout<<setfill(' ');
196     cout<<setw(20)<<left<<"Ma Kieu Phong"<<setw(20)<<left<<"Nguoi Thue";
197     cout<<setw(20)<< left<<"Kieu Phong"<<setw(20)<<left<<"So Ngay Thue"<<setw(20)<<left<<"So Lu";
198     cout<<setw(20)<<left<<"Gia Thue";
199     cout<<endl;
200     cout<<setfill('-');
201     cout<<setw(120)<<'-'<<endl;
202     cout<<setfill(' ');
203 }
204
205 static void InHoaDon() {
206     cout<<setfill('-');

```

```

207     cout<<setw(120)<<"-"<<endl;
208     cout<<setfill(' ');
209     cout<<setw(65)<<right<<"HOA DON"<<endl;
210     cout<<setfill("-");
211     cout<<setw(120)<<"-"<<endl;
212     cout<<setfill(' ');
213     cout<<setw(17)<<left<<"Ma Khach Hang"<<setw(17)<<left<<"Ho Ten";
214     cout<<setw(17)<< left<<"Kieu Phong"<<setw(17)<<left<<"So Ngay Thue"<<setw(17)<<left<<"So Lu";
215     cout<<setw(17)<<left<<"Gia Mot Phong"<<setw(17)<<left<<"Tong";
216     cout<<endl;
217     cout<<setfill("-");
218     cout<<setw(120)<<"-"<<endl;
219     cout<<setfill(' ');
220 }
221
222 //So sanh theo loai phong
223 static bool SapXepLoaiPhong(Phong a, Phong b){return a.getkieuPhong() < b.getkieuPhong();}
224
225 //So sanh theo so phong con lai co the cho thue
226 static bool SapXepSoPhong(Phong a, Phong b){return a.getsoPhong() < b.getsoPhong();}
227
228
229
230 int main(){
231     //Nhap va in danh sach phong
232     int slPhong;
233     cout<<"Nhap so luong phong cho thue: ", cin>>slPhong;
234     vector<Phong> DSP(slPhong);
235     cin.ignore();
236     for(int i=0;i<slPhong;i++){
237         cout<<"Nhap thong tin cho phong thu "<<i+1<<" : "<<endl;
238         DSP[i].NhapPhong();
239     }
240     if(!DSP.empty()){
241         InPhong();
242         for(int i=0;i<DSP.size();i++){
243             DSP[i].InPhong();
244         }
245     }
246
247     //Nhap va in danh sach khach hang
248     int slKH;
249     cout<<"\nNhap so khach hang: ", cin>>slKH;
250     vector<KhachHang> DSKH(slKH);
251     cin.ignore();
252     for(int i=0;i<slKH;i++){
253         cout<<"\nNhap thong tin cho khach hang thu "<<i+1<<" : "<<endl;
254         DSKH[i].NhapKH();
255     }
256     if(!DSKH.empty()){
257         InKhachHang();
258         for(int i=0;i<DSKH.size();i++){

```

```

259         DSKH[i].InKH();
260     }
261 }
262
263 //Nhap va in danh sach cho khach hang thue phong
264 vector<BangSapXep> BSX;
265 for(int i=0;i<slKH;i++){
266     while(DSKH[i].getsoPhong(>0){
267         cout<<"\nChon phong cho khach hang: "<<DSKH[i].getHoTen()<<endl;
268         for(int j=0;j<DSP.size();j++){
269             cout << "Phong so " << DSP[j].getmakieuPhong() << ", kieu " <<
270 DSP[j].getkieuPhong();
271             cout << ", so phong con lai: " << DSP[j].getsoPhong() << endl;
272         }
273         int ma;
274         cout<<"Nhap ma kieu phong muon thue: ", cin>>ma;
275         for(int k=0;k<DSP.size();k++){
276             if(ma==DSP[k].getmakieuPhong()){
277                 if(DSP[k].getsoPhong(>0){
278                     cout<<"Con " <<DSP[k].getsoPhong()<<" phong voi ma ki
279 " <<DSP[k].getmakieuPhong()<<" co the thue!"<<endl;
280                     BangSapXep NhanPhong;
281                     int songayThue_,soluongThue_;
282                     cout<<"Nhap so luong phong can thue: ", cin>>soluongThue_;
283                     while(DSP[k].getsoPhong()-soluongThue_<0 ||
284 DSKH[i].getsoPhong()-soluongThue_<0){
285                         cout<<"\nSo luong phong can thue khong hop le
286                         cout<<"Vui long nhap lai: ", cin>>soluongThue_
287                     };
288                     cout<<"So ngay thue: ", cin>>songayThue_;
289                     while(songayThue_<=0){
290                         cout<<"\nSo ngay thue khong hop le!"<<endl;
291                         cout<<"Vui long nhap lai: ", cin>>songayThue_
292                     }
293                     NhanPhong.NhapP(DSKH[i],DSP[k],songayThue_,soluongThue_);
294                     DSP[k].giamsoPhong(soluongThue_);
295                     DSKH[i].giamsoPhong(soluongThue_);
296                     BSX.push_back(NhanPhong);
297                     break;
298                 }
299                 else if(DSP[k].getsoPhong()==0) cout<<"Da het phong can thue
300 long chon phong khac!"<<endl;
301             }
302             else {
303                 if(k==DSP.size()-1 && ma!=DSP[k].getmakieuPhong()){
304                     cout<<"Khong tim thay kieu ma phong nay!\n"<<"Vui long nhap
305                     break;
306                 }
307             }
308         }
309     }
310 }

```

```

311         if(!BSX.empty()){
312             InPhongDaDat();
313             for(int i=0;i<BSX.size();i++){
314                 BSX[i].InBang();
315             }
316         }
317
318         //Sap xep theo loai phong
319         cout<<"\nSap xep danh sach sap xep phong theo loai phong:"<<endl;
320         sort(DSP.begin(),DSP.end(),SapXepLoaiPhong);
321         InPhong();
322         for(auto sx : DSP){
323             sx.InPhong();
324         }
325
326         //Sap xep theo so luong phong con lai co the cho thue
327         cout<<"\nSap xep danh sach sap xep phong theo so phong con lai:"<<endl;
328         sort(DSP.begin(),DSP.end(),SapXepSoPhong);
329         InPhong();
330         for(auto sx : DSP){
331             sx.InPhong();
332         }
333
334         //In hoa don cho khách hàng
335         cout<<"\nLap hoa don cho moi khách hàng:"<<endl;
336         InHoaDon();
337         for(auto kh : BSX){
338             kh.XuatHoaDon();
339         }
340         return 0;
341     }

```

Kết quả:

#Nhập và in danh sách khách hàng:

Nhap so khach hang: 2

Nhap thong tin cho khach hang thu 1:

Ho va ten: Doanh

Dia chi: Ha Noi

So dien thoai: 12345

So phong can thue: 2

Loai phong: vip

Nhap thong tin cho khach hang thu 2:

Ho va ten: Dan

Dia chi: Ha Noi

So dien thoai: 12346

So phong can thue: 1

Loai phong: don

DANH SACH KHACH HANG

Ma Khach Hang	Ho Va Ten	Dia Chi	So Dien Thoai	So Phong Can Thue	Loai Phong
100	Doanh	Ha Noi	12345	2	vip
101	Dan	Ha Noi	12346	1	don

#Nhập và in danh sách phòng cho thuê:

Nhap so luong phong cho thue: 3

Nhap thong tin cho phong thu 1:

Kieu phong(don, doi, vip): doi

So phong co the cho thue: 14

Tien thue phong: 50000

Nhap thong tin cho phong thu 2:

Kieu phong(don, doi, vip): vip

So phong co the cho thue: 10

Tien thue phong: 95000

Nhap thong tin cho phong thu 3:

Kieu phong(don, doi, vip): don

So phong co the cho thue: 6

Tien thue phong: 38000

DANH SACH PHONG

Ma Kieu Phong	Kieu Phong	So Phong Co The Cho Thue	Gia Thue Phong
100	doi	14	50000
101	vip	10	95000
102	don	6	38000

Nhập danh và in sách sắp xếp phòng cho mỗi khách hàng

```

Chon phong cho khach hang: Doanh
Phong so 100, kieu doi, so phong con lai: 14
Phong so 101, kieu vip, so phong con lai: 10
Phong so 102, kieu don, so phong con lai: 6
Nhap ma kieu phong muon thue: 100
Con 14 phong voi ma kieu 100 co the thue!
Nhap so luong phong can thue: 2
So ngay thue: 365

```

```

Chon phong cho khach hang: Dan
Phong so 100, kieu doi, so phong con lai: 12
Phong so 101, kieu vip, so phong con lai: 10
Phong so 102, kieu don, so phong con lai: 6
Nhap ma kieu phong muon thue: 102
Con 6 phong voi ma kieu 102 co the thue!
Nhap so luong phong can thue: 1
So ngay thue: 90

```

DANH SACH DAT PHONG					
Ma Kieu Phong	Nguyen Thue	Kieu Phong	So Ngay Thue	So Luong	Gia Thue
100	Doanh	doi	365	2	50000
102	Dan	don	90	1	38000

#Sắp xếp danh sách phòng theo loại phòng:

Sap xep danh sach sap xep phong theo loai phong:

DANH SACH PHONG			
Ma Kieu Phong	Kieu Phong	So Phong Co The Cho Thue	Gia Thue Phong
100	doi	14	50000
102	don	5	38000
101	vip	8	95000

#Sắp xếp danh sách phòng theo số phòng còn lại(tăng dần):

Sap xep danh sach sap xep phong theo so phong con lai:

DANH SACH PHONG			
Ma Kieu Phong	Kieu Phong	So Phong Co The Cho Thue	Gia Thue Phong
102	don	5	38000
101	vip	8	95000
100	doi	14	50000

Lập hóa đơn cho mỗi khách hàng:

Lap hoa don cho moi khach hang:

HOA DON						
Ma Khach Hang	Ho Ten	Kieu Phong	So Ngay Thue	So Luong	Gia Mot Phong	Tong
100	Doanh	doi	365	2	50000	100000
101	Dan	don	90	1	38000	38000