

# Hệ thống quản lý đặt phòng của công ty du lịch

Một công ty du lịch hỗ trợ khách hàng đặt trước phòng của các khách sạn khác nhau tại Hà nội. Khách hàng khi book phòng chỉ cần cung cấp thông tin gồm

- **Mã khách sạn:** mã theo quy định trên hệ thống của công ty, và chỉ gồm 3 ký tự viết tắt
- Tổng số người
- **Mã phòng** trong hệ thống của khách sạn
- **Ngày/tháng/năm** mong muốn nhận phòng (đọc theo dạng xâu ký tự)
- Số lượng trẻ em đi kèm (dưới 6 tuổi) trong tổng số người ở trên

Các thông tin này sẽ được hệ thống ghi vào 1 file log (dạng file văn bản) để đẩy lên server xử lý tiếp. Thông tin mỗi yêu cầu của khách sẽ được lưu trên 1 dòng. Định dạng file log là

- Dòng đầu tiên là số lượng yêu cầu trong file
- Các dòng tiếp theo mỗi dòng lần lượt sẽ là **mãYêuCầu sốNgười sốPhòng mãKháchSạn DD-MM-YYYY kids** trong đó **mãYêuCầu** sẽ là mã tự sinh của hệ thống để phân biệt các yêu cầu của khách và KHÔNG bị trùng nhau. Các thông tin này được cách nhau bởi 1 dấu cách trống.

Ví dụ

```
6
HUA2345 3 698 PCS 11-05-2023 2
HUA1234 4 698 PCS 11-05-2023 2
JHF3453 1 595 KIQ 06-08-2024 0
PCM2345 3 664 PPO 18-03-2026 0
MEN2345 1 195 TKQ 27-08-2026 0
SCS3456 3 593 MEH 10-07-2026 2
```

File log này chứa 6 yêu cầu

với yêu cầu: **HUA2345 3 698 PCS 11-05-2023 2**

- Mã yêu cầu là **HUA2345** (mã này luôn có độ dài cố định là 7 gồm cả chữ cái và chữ số).
- Số lượng người là 3
- Số phòng tại khách sạn là 698
- Mã khách sạn là **PCS**
- Ngày tháng năm lần lượt là 11-05-2023 (tức là ngày 11 tháng 5 năm 2023 theo định dạng **DD-MM-YYYY**)
- Trong số lượng 3 người thì có 2 trẻ em (tức là 1 người lớn kèm 2 trẻ em)

Hãy viết chương trình để thực hiện các công việc sau

1. **Đọc file log nội dung các yêu cầu**, tên file đầu vào sẽ là "booking\_orders.txt", lưu vào mảng động hoặc danh sách liên kết. Khi nạp xong hãy in ra số lượng yêu cầu đọc được.
2. Theo quy định chung của mỗi khách sạn: Nếu có trẻ em thì phải ít nhất phải có 1 người lớn đi cùng. Hãy Lọc trong danh sách đầu vào và **in ra số lượng các yêu cầu KHÔNG** hợp lệ theo tiêu chí trên.
3. Để hỗ trợ cho việc tra cứu thông tin chi tiết yêu cầu đặt phòng, hãy lưu lại danh sách order trên vào cây tìm kiếm nhị phân hoặc bảng băm với **khóa tìm kiếm là mã yêu cầu**, và thực hiện các chức năng ở dưới
  - a) **Thực hiện tìm kiếm thông tin** 1 yêu cầu theo mã yêu cầu nhập từ bàn phím. Nếu có thì in ra đủ thông tin chi tiết của yêu cầu đặt phòng đó (ngày-tháng-năm, số lượng người,...). Ngược lại in ra thông báo là "Ma yeu cau khong dung!"
  - b) **Thực hiện cập nhật yêu cầu** (chỉ cho phép thay đổi ngày/tháng/năm, số lượng người, và số lượng trẻ em đi cùng). Cần check yêu cầu cập nhật này thỏa mãn ràng buộc về số lượng trẻ em.  
gợi ý: Ngày-tháng-năm nên đọc theo định dạng xâu ký tự giống file đầu vào.  
Nếu cập nhật thành công cần in ra thông báo "Cap nhat yeu cau thanh cong", ngược lại in ra thông báo

“Cap nhat khong thanh cong” nếu dữ liệu không đúng, hoặc “Ma yeu cau khong dung” nếu mã yêu cầu sai.

- c) **Thực hiện hủy yêu cầu**, hãy nhập mã yêu cầu của khách sạn nào được yêu cầu nhiều nhất, và thực hiện việc xóa khỏi cây hoặc bảng băm trên (nếu mã yêu cầu đó tồn tại). Khi hủy thành công cần in ra thông báo “Huy thanh cong”, ngược lại in ra thông báo “Huy khong thanh cong”

4. **Ghi thông tin các yêu cầu hiện tại trong cây/bảng băm ra file** “booking\_output.txt” theo format các dòng và trường như file đầu vào. Sau khi ghi file xong, hãy in ra màn hình số lượng mã yêu cầu đã ghi được vào file.

#### Câu hỏi phụ (Không bắt buộc)

- (OP1) Công ty muốn tìm xem **mã khách sạn nào được đăng ký thuê phòng nhiều nhất**. Hãy duyệt các yêu cầu hợp lệ trên cây/bảng băm, tìm và in ra xem mã khách sạn nào được yêu cầu nhiều nhất, và số lượng yêu cầu.
- (OP2) **Các yêu cầu hợp lệ về thời gian** là các yêu cầu có ngày đặt phòng phải sau ngày hiện tại (ngày hiện tại có thể nhập từ bàn phím hoặc lấy ngày hệ thống). Hãy lọc trong danh sách/mảng đầu vào và in ra màn hình các mã yêu cầu KHÔNG hợp lệ về ngày tháng.
- (OP3) Giả sử công ty chỉ cho phép **thời gian đặt phòng tối đa trước 2 tháng (60 ngày)** từ ngày hiện tại (lấy của hệ thống hoặc nhập vào). Hãy lọc trong danh sách/mảng đầu vào và in ra màn hình các mã yêu cầu không hợp lệ do thời gian đặt phòng quá xa (so với ngày hiện tại) nếu có

Khai báo CTDL để lưu trữ danh sách booking	1d
Xây dựng chương trình với menu lựa chọn	1d
Đọc dữ liệu vào từ file <b>Thiếu in ra số lượng trừ 0.25</b>	1d
Đếm số lượng yêu cầu không hợp lệ	1d
Xây dựng cây tìm kiếm nhị phân hoặc bảng băm để tra cứu yêu cầu Nếu sinh viên dừng ở bước này mà có 1 số thao tác cơ bản của CTDL cây hoặc bảng băm có thể chiếu cố cho 0.25 – 0.5 điểm	1d
Thực hiện tìm kiếm <b>Thiếu in ra Không tìm thấy trừ 0.5</b>	1d
Thực hiện hủy yêu cầu <b>Trừ 0.25 nếu in thiếu thông báo trong trường hợp yêu cầu không tồn tại</b> <b>Trừ 0.25 nếu in thiếu thông báo trong trường hợp dữ liệu không hợp lệ</b>	1d
Thực hiện cập nhật <b>Nếu không check điều kiện hợp lệ của yêu cầu trừ 0.25</b> <b>Trừ 0.25 nếu không in ra thông báo</b>	1d
Ghi thông tin đang có ra file <b>Nếu không ghi số lượng trừ 0.5</b>	1d
Giải phóng các cấu trúc dữ liệu cấp phát động ở trên	1d
Thống kê mã khách sạn được đăng ký thuê nhiều nhất <b>Không in ra số lượng trừ 0.5</b>	1d
Lọc các yêu cầu hợp lệ về thời gian	1d (câu cộng điểm)
Lọc các yêu cầu quá sớm	1d (câu cộng điểm)

#### Menu lựa chọn dạng

Chương trình quản lý đặt phòng  
Vui lòng chọn một trong các chức năng sau  
[1] Đọc dữ liệu từ file log  
[2] Kiểm tra số lượng yêu cầu không hợp lệ  
[3] Tra cứu theo mã yêu cầu  
[4] Thay đổi yêu cầu

[5] Huy yeu cau  
[6] Luu danh sach yeu cau ra file  
[7] Thong ke khach san duoc ua thich nhat  
Lua chon cua bạn là:

### Gợi ý:

Nếu muốn lấy ngày hiện tại của hệ thống bạn có thể dùng code sau

```
#include <stdio.h>
#include <time.h>

void printCurrDate()
{
    time_t t = time(NULL);
    struct tm tm = *localtime(&t);
    printf("Hien tai la: %d-%02d-%02d\n", tm.tm_year + 1900, tm.tm_mon + 1, tm.tm_mday);
}
```

Để chuyển chuỗi ký tự thành ngày tháng năm bạn có thể dùng

```
#include <stdio.h>
#include <stdlib.h>
#include <time.h>

int main() {
    char datetime_string[] = "02-03-2023";
    struct tm datetime_tm = {0};

    // Parse datetime string into tm structure
    if (strptime(datetime_string, "%d-%m-%Y", &datetime_tm) == NULL) {
        fprintf(stderr, "Failed to parse datetime string.\n");
        exit(EXIT_FAILURE);
    }
    printf("Date %04d-%02d-%02d\n", datetime_tm.tm_year + 1900, datetime_tm.tm_mon + 1,
datetime_tm.tm_mday);
    return 0;
}
```