**TRƯỜNG ĐẠI HỌC BÁCH KHOA**

**KHOA CÔNG NGHỆ THÔNG TIN**



**ĐỒ ÁN LẬP TRÌNH TÍNH TOÁN**

**TÊN ĐỀ TÀI: XÂY DỰNG ỨNG DỤNG ĐẶT CHỖ NGỒI TRÊN MÁY BAY CỦA HÃNG HÀNG KHÔNG ITF-AIRWAY**

Người hướng dẫn**: GVC. ThS. TRẦN HỒ THỦY TIÊN**

Sinh viên thực hiện**:**

**Lê Ngọc Hưng LỚP: 20T1 NHÓM: 20NH10**

**Nguyễn Hữu Đức LỚP: 20T2 NHÓM: 20NH10**

**Đà Nẵng, 08/2021**

MỤC LỤC

[MỤC LỤC 2](#_Toc72349061)

[DANH MỤC HÌNH VẼ 3](#_Toc72349062)

[MỞ ĐẦU 5](#_Toc72349063)

[1. TỔNG QUAN ĐỀ TÀI 7](#_Toc72349064)

[2. CƠ SỞ LÝ THUYẾT 7](#_Toc72349065)

[2.1. Ý tưởng 7](#_Toc72349066)

[2.2. Cơ sở lý thuyết 7](#_Toc72349067)

[3. TỔ CHỨC CẤU TRÚC DỮ LIỆU VÀ THUẬT TOÁN 8](#_Toc72349068)

[3.1. Phát biểu bài toán 8](#_Toc72349069)

[3.2. Cấu trúc dữ liệu 8](#_Toc72349070)

[3.3. Thuật toán 14](#_Toc72349071)

[4. CHƯƠNG TRÌNH VÀ KẾT QUẢ 19](#_Toc72349072)

[4.1. Tổ chức chương trình 19](#_Toc72349073)

[4.2. Ngôn ngữ cài đặt 20](#_Toc72349074)

[4.3. Kết quả 22](#_Toc72349075)

[4.3.1. Giao diện chính của chương trình 22](#_Toc72349076)

[4.3.2. Kết quả thực thi của chương trình 24](#_Toc72349077)

[4.3.3. Nhận xét đánh giá 29](#_Toc72349078)

[5. KẾT LUẬN VÀ HƯỚNG PHÁT TRIỂN 30](#_Toc72349079)

[5.1. Kết luận 30](#_Toc72349080)

[5.2. Hướng phát triển 30](#_Toc72349081)

[TÀI LIỆU THAM KHẢO 31](#_Toc72349082)

[PHỤ LỤC 32](#_Toc72349083)

DANH MỤC HÌNH VẼ

[Hình 1. Các thao tác với File. 9](#_Toc73523325)

[Hình 2. Sơ đồ cấu trúc chương trình. 15](#_Toc73523326)

[Hình 3. Sơ đồ khối hàm Open. 15](#_Toc73523327)

[Hình 4. Sơ đồ khối hàm Menu 16](#_Toc73523328)

[Hình 5. Sơ đồ khối hàm Seats 17](#_Toc73523329)

[Hình 6. Sơ đồ khối làm updateSeat. 17](#_Toc73523330)

[Hình 7. Sơ đồ khối hàm Airline. 18](#_Toc73523331)

[Hình 8. Sơ đồ khối hàm Check. 19](#_Toc73523332)

[Hình 9. Giao diện mở đầu chương trình 22](#_Toc73523333)

[Hình 10. Nhân viên nhập sai mã Opening Code 22](#_Toc73523334)

[Hình 11. Nhân viên nhập sai mã Flight Code. 22](#_Toc73523335)

[Hình 12. Nhân viên nhập đúng mã Opening Code và Flight Code. 23](#_Toc73523336)

[Hình 13. Hiển thị sơ đồ chỗ ngồi. 24](#_Toc73523338)

[Hình 14. Tiến hành đặt chỗ 25](#_Toc73523339)

[Hình 15. Nhập sai chỗ 26](#_Toc73523340)

[Hình 16. Đặt chỗ thành công 26](#_Toc73523341)

[Hình 17. Thoát khỏi chương trình 27](#_Toc73523342)

[Hình 18.File ánh xạ chỗ ngồi sau khi đặt chỗ 28](#_Toc73523343)

[Hình 19. File ánh xạ mã vé tương ứng với số chỗ. 29](#_Toc73523344)

[Hình 20. Code hàm Intro. 32](#_Toc73523345)

[Hình 21. Code hàm read 32](#_Toc73523346)

[Hình 22. Code hàm open. 32](#_Toc73523347)

[Hình 23. Code hàm menu 33](#_Toc73523348)

[Hình 24. Code hàm seats. 33](#_Toc73523349)

[Hình 25. Code hàm updateSeat. 34](#_Toc73523350)

Hình 26. Code hàm airline…………………………………………………….34

Hình 27. Code hàm check……………………………………………………..35

Hình 28. Code hàm updateFile………………………………………………..35

LỜI MỞ ĐẦU

**1. Mục đích thực hiện đề tài**

**-** Ứng dụng các kiến thức đã học trong các học phần trước để xây dựng ứng dụng giải quyết một vấn đề trong thực tiễn. Cụ thể là xây dựng ứng dụng đặt chỗ ngồi trên máy bay của hãng hàng không ITF-Airway.

- Phát triển kỹ năng tư duy, cách xây dựng thuật toán.

- Nâng cao một số kỹ năng mềm như làm việc nhóm, thuyết trình , giải quyết vấn đề

**2. Mục tiêu đề tài**

* Viết chương trình bằng C/C++ để giải quyết bài toán.
* Viết các hàm để xây dựng chức năng.
* Sử dụng các mảng song song để lưu dữ liệu quan hệ.
* Sử dụng các hàm làm việc với file.
* Sử dụng các hàm xử lý chuỗi.

**3. Phạm vi và đối tượng nghiên cứu**

* Tìm hiểu cấu trúc dữ liệu, làm việc với file, kiểu mảng và các kiến thức liên quan để áp dụng vào việc viết chương trình.
* Liên kết các kiến thức đã học từ các học phần để xây dựng thuật toán và cách giải quyết bài toán phù hợp, tối ưu.

**4. Phương pháp nghiên cứu**

* Làm việc nhóm, viết báo cáo.
* Phân tích vấn đề, từ đó tìm ra các hướng giải quyết và chọn cách tối ưu nhất để tiến hành viết chương trình.
* Tìm hiểu các nguồn tài liệu về lập trình C/C++, làm việc với file., mảng…
* Chia vấn đề thành các bài toán nhỏ hơn để giải quyết.

**5. Cấu trúc của đồ án môn học:**

* Trang bìa.
* Trang bìa trong.
* Trang Mục lục.
* Danh mục hình vẽ.
* Trang “ Lời mở đầu ”.
* Tổng quan đề tài.
* Cơ sở lý thuyết.

+ Ý tưởng.

+ Cơ sở lý thuyết.

* Tổ chức cấu trúc dữ liệu và thuật toán

+ Phát biểu bài toán.

+ Cấu trúc dữ liệu.

+ Thuật toán.

* Chương trình và kết quả

+ Tổ chức chương trình.

+ Ngôn ngữ cài đặt.

* Kết quả.
* Kết luận và hướng phát triển ,
* Trang Tài liệu tham khảo.
* Các trang Phụ lục.

1. TỔNG QUAN ĐỀ TÀI

* Hãng hàng không ITF-Airway điều hành một chiếc máy bay với số chỗ ngồi MaxSeats.
* Hãng hàng không này cần xây dựng chương trình để xác định ghế nào còn trống để xếp chỗ cho khách hàng trên mỗi chuyến bay.
* Sơ đồ chỗ ngồi trên máy bay được mô tả trên một file.

1. CƠ SỞ LÝ THUYẾT
   1. Ý tưởng

* Dựa vào các kiến thức về hàm, mảng, làm việc với file để xây dụng thuật toán, viết chương trình.
* Dùng một file seats.txt để xếp chỗ ngồi ban đầu trên máy bay gồm có 260 chỗ và được quy ước thành một mảng 2 chiều. Cột đầu tiên được hiểu là kí tự hàng chỗ ngồi từ A-Z(gồm 26 hàng). Cột thứ 2-10 là vị trí chỗ ngồi.
* Vị trí được xem là một xâu kí tự có độ dài là 2 và hợp lệ nếu kí tự đầu tiên bắt đầu từ A-Z và kí tự thứ 2 bắt đầu từ 1-9.
* Kết thúc chương trình, thu được file output.txt là file mã vé tương ứng với chỗ ngồi đã đặt.
  1. Cơ sở lý thuyết
* Dùng các thư viện trong lập trình C/C++ để hỗ trợ cho các chức năng chính trong chương trình:
* <iostream> -- Định nghĩa các đối tượng để đọc và viết ra các dòng tiêu chuẩn (standard stream). Đây thường là tiêu đề duy nhất cần phải bao gồm để tiến hành xuất và nhập (dữ liệu) từ C++.
* <windows.h> là một header của Windows dành riêng cho ngôn ngữ lập trình C và C++. Trong đó chứa các khai báo cho tất cả các hàm (function) trong Windows API, tất cả các macro thường dùng bởi các lập trình viên Windows, và tất cả các kiểu dữ liệu (data type) sử dụng cho nhiều hàm và hệ thống con (subsystem).
* <string.h> trong Thư viện C++ chuẩn định nghĩa một kiểu biến, một macro và các hàm đa dạng để thao tác các mảng ký tự.

1. TỔ CHỨC CẤU TRÚC DỮ LIỆU VÀ THUẬT TOÁN
   1. Phát biểu bài toán

* Input:
* File chỗ ngồi trên máy bay.
* Mã vé của hành khách đi máy bay.
* Output: File ánh xạ thông tin Mã vé của hành khách tương ứng chỗ ngồi trên máy bay.
  1. Cấu trúc dữ liệu
* Thư viện sử dụng:

#include<iostream>

#include<windows.h>

#include<string.h>

* File:

+ Chỗ ngồi trên máy bay: seats.txt

+ Ánh xạ thông tin Mã vé của hành khách đi máy bay: output.txt

* Kiểu chuỗi kí tự: string flightInfo để lưu mã chuyến bay
* Kiểu mảng 2 chiều để lưu sơ đồ chỗ ngồi và mã vé tương ứng với vị trí ghế ngồi đã đặt: char arr[26][10], char output[26][10].

**3.2.1 Làm việc với File**

* Tại sao chúng ta cần đến file?
* Dữ liệu được lưu ở biến của chương trình, và nó sẽ biến mất khi chương trình kết thúc. Sử dụng file để lưu trữ dữ liệu cần thiết để đảm bảo dữ liệu của chúng ta không bị mất ngay cả khi chương trình của chúng ta ngừng chạy.
* Nếu chương trình của bạn có đầu vào(input) là lớn, bạn sẽ rất vất vả nếu phải nhập mỗi khi chạy. Thay vào đó, hãy lưu vào file và chương trình của bạn sẽ tự đọc mỗi lần khởi chạy.
* Dễ dàng sao chép, di chuyển dữ liệu giữa các thiết bị với nhau.
* Các kiểu file:
* File văn bản – text file
* File nhị phân – binary file
* Các thao tác với file:

Khi làm việc với file, ta cần khai báo 1 con trỏ kiểu FILE. Việc khai báo này là cần thiết để có sự kết nối giữa chương trình và tập tin cần thao tác.

FILE \*fptr;

* Thao tác mở file:

Để đọc ghi file trong C cũng như trong mọi ngôn ngữ lập trình, việc đầu tiên cần làm là mở file mà ta muốn làm việc. Trong ngôn ngữ lập trình C/C++, chúng ta có thể mở file bằng cách sử dụng hàm fopen() như sau:

fptr = fopen("fileopen","mode")

Trong đó mode là một tham số chúng ta cần chỉ định.

Dưới đây là các giá trị có thể có của tham số mode nói trên:

Graphical user interface, text, email

Description automatically generated

Hình 1. Các thao tác với File

* Đọc file trong C/C++:

Dưới đây là hàm đơn giản nhất để đọc một ký tự riêng rẽ từ file:

int fgetc( FILE \* fptr);

Hàm fgetc() đọc một ký tự từ một file tham chiếu bởi con trở fp. Giá trị trả về là ký tự đọc được nếu thành công, và trong trường hợp lỗi trả về EOF. Hàm dưới đây cho phép bạn đọc chuỗi từ một luồng:

char \*fgets( char \*buf, int n, FILE \*fp );

Hàm fgets() đọc n-1 ký tự từ một luồng vào tham chiếu bởi fp. Nó copy chuỗi đọc đến bộ đệm buf, gán ký tự null vào kết thúc chuỗi.Nếu hàm gặp phải một ký tự newline (dòng mới) (xuống dòng) '\n' hoặc ký tự EOF trước khi đọc được số lượng tối đa các ký tự, nó sẽ chỉ trả về các ký tự cho đến ký tự xuống dòng và ký tự xuống dòng mới. Bạn có thể sử dụng hàm int fscanf(FILE \*fp, const char \*format, ...) để đọc chuỗi từ một file, nhưng dừng việc đọc ở khoảng trắng đầu tiên gặp phải:

Để làm việc với file văn bản, chúng ta sẽ sử dụng fprintf() và fscanf().

**3.2.2 Kiểu chuỗi**

* Nhập xuất chuỗi:
* Nhập chuỗi từ bàn phím

Để nhập một chuỗi ký tự từ bàn phím, ta sử dụng hàm gets()

Cú pháp: gets(<Biến chuỗi>)

Ví dụ:

char Ten[20];

gets(Ten);

Ta cũng có thể sử dụng hàm scanf() để nhập dữ liệu cho biến chuỗi, tuy nhiên lúc này ta chỉ có thể nhập được một chuỗi không có dấu khoảng trắng. Ngoài ra, hàm gets() (trong stdio.h) cũng được sử dụng để nhập chuỗi.

* Xuất chuỗi lên màn hình

Để xuất một chuỗi (biểu thức chuỗi) lên màn hình, ta sử dụng hàm puts().

Cú pháp: puts(<Biểu thức chuỗi>)

Ví dụ: Nhập vào một chuỗi và hiển thị trên màn hình chuỗi vừa nhập.

#include<conio.h>

#include<stdio.h>

#include<string.h>

int main()

{

char Ten[12];

char queQuan[15];

printf("Nhap Ten: ");fflush(stdin);gets(Ten);

printf("Nhap Que Quan: ");fflush(stdin);gets(queQuan);

printf("Chuoi vua nhap: ");puts(Ten);puts(queQuan);

getch();

return 0;

}

Ngoài ra, ta có thể sử dụng hàm printf(), puts() (trong conio.h) để hiển thị chuỗi lên màn hình.

* Một số hàm xử lý chuỗi (trong string.h)
* Cộng chuỗi - Hàm strcat()

Cú pháp: char \*strcat(char \*des, const char \*source)

Hàm này có tác dụng ghép chuỗi nguồn vào chuỗi đích.

Ví dụ: Nhập vào họ lót và tên của một người, sau đó in cả họ và tên của họ lên màn hình.

#include<conio.h>

#include<stdio.h>

#include<string.h>

int main()

{

char HoLot[30], Ten[12];

printf("Nhap Ho Lot: ");gets(HoLot);

printf("Nhap Ten: ");gets(Ten);

strcat(HoLot,Ten); /\* Ghep Ten vao HoLot\*/

printf("Ho ten la: ");puts(HoLot);

getch();

return 0;

}

* Xác định độ dài chuỗi - Hàm strlen()

Cú pháp: int strlen(const char\* s)

Ví dụ: Sử dụng hàm strlen xác định độ dài một chuỗi nhập từ bàn phím.

#include<conio.h>

#include<stdio.h>

#include<string.h>

int main(){

char Chuoi[1000];

int Dodai;

printf("Nhap chuoi: ");gets(Chuoi);

Dodai = strlen(Chuoi);

printf("Chuoi vua nhap: ");puts(Chuoi);

printf("Co do dai %d",Dodai);

getch();

return 0;

}

**3.2.3 Kiểu mảng**

* Mảng tĩnh:
* Khai báo mảng tĩnh

Trong C, mảng được dùng để biểu thị một cấu trúc của một dãy nhiều giá trị có cùng một kiểu được xếp thứ tự. Một mảng gọi là tĩnh nếu độ dài của dãy mảng này cố định. Sự khai báo của mảng tĩnh có cú pháp sau:

int array[n];

trong đó, tên của mảng là array sẽ có thể chứa được n giá trị của kiểu cơ bản int. Trong thực hành, phần bộ nhớ cho n giá trị nguyên này được để dành riêng và được gán cho mảng này (mặc dù giá trị của các phần tử trong mảng chưa được xác định). Biến array thực chất là một kiểu tham chiếu của kiểu nguyên; nó khởi thủy sẽ chỉ tới địa chỉ của giá trị đầu tiên trong mảng.

* Truy cập các phần tử

Các giá trị của một mảng được gọi là các phần tử trong mảng.Truy cập đến các phần tử này là dùng cặp kí tự ngoặc vuông dạng [k]. Trong đó k là chỉ số (hay vị trí thứ tự đếm từ 0). Như vậy, phần tử thứ k trong mảng array sẽ có cú pháp:

array[k]

Giá trị trả về của array[k] chính là giá trị mà nó chứa ở vị trí k. Thoạt nhìn thì cú pháp của việc truy cập này trông giống như cú pháp khi mảng array được khai báo nhưng về chức năng thì hoàn toàn khác nhau.Chỉ số bắt đầu của một mảng là 0. Như vậy, chỉ số lớn nhất của một mảng bằng tổng số các phần tử trong mảng trừ đi 1. Thí dụ mảng A có 10 phần tử thì giá trị của phần tử đầu tiên của A là A[0] và của phần tử cuối dùng là A[9].

Một cách truy cập khác là dùng con trỏ số học để tham chiếu đến giá trị của các phần tử trong mảng.Bảng sau đây sẽ minh họa cách dùng của cả hai phương pháp:

Array Chỉ số và con trỏ số học

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Phần tử vị trí | 0 | 1 | 2 | n |
| Kiểu dùng cơ bản | array[0] | array[1] | array[2] | array[n] |
| Dùng con trỏ | \*array | \*(array+1) | \*(array+2) | \*(array+n) |

* Các mảng động

C không cung cấp phương tiện để kiểm tra biên cho các mảng. Nghĩa là nó không thể bắt được các lỗi khi gán cho một mảng chỉ số âm hay chỉ số vượt quá độ đài của mảng đó. Và hơn thế nữa các chỉ số trong một mảng có thể vượt khỏi độ dài sẵn có của mảng đó.Vì các mảng là thuần nhất, tức là nó chỉ chứa các dữ liệu có chung một kiểu nên hai thành phần thông tin cần nhớ là địa chỉ của phần tử đầu tiên và kiểu của dữ liệu.

* Các mảng đa chiều

C có hỗ trợ việc dùng mảng đa chiều. Việc định nghĩa chúng giống như là tạo ra mảng của các mảng , mặc dù vậy trong thực tế nó không hoàn toàn đúng. Cú pháp sau:

int array2D[số\_hàng][số\_cột];

sẽ định nghĩa một mảng hai chiều; chiều thứ nhất có số\_hàng phần tử. Chiều thứ hai sẽ có số\_hàng \* số\_cột các phần tử -- một tập hợp của số\_cột các phần tử mà mỗi phần tử là một chiều thứ nhất. Các mảng đa chiều hoàn toàn có thể được xem như là dãy của các con trỏ. Trong thí dụ trên, array2D (nếu số\_hàng là 1) sẽ là một tham chiếu giá trị nguyên mà nó chỉ tới một mảng của số\_cột các phần tử.

* 1. Thuật toán

Trước giờ Checkin, nhân viên nhập vào mã open để chương trình hiển thị.

**Bước 1:** Hiển thị thông tin chuyến bay.

**Bước 2:** Mời đặt chỗ: Nhập vào số chỗ >0.

* Lặp nhập mã vé tương ứng với số chỗ.
* Tìm số ghế trống tương ứng với số chỗ nhập vào.
* Hiển thị số ghế trống tìm thấy.
* Chọn số ghế tìm thấy.
* Lưu mã vé tương ứng với số ghế tìm thấy vào file ánh xạ output.txt
* Quay lại bước 2 cho đến khi hết chỗ hoặc nhân viên nhập vào mã close.

**Bước 3:** Hiển thị file ánh xạ.

Sơ đồ cấu trúc chương trình được mô tả như sau:

Diagram

Description automatically generated

Hình 2. Sơ đồ cấu trúc chương trình

Sơ đồ khối các hàm:

* Hàm open:

Diagram

Description automatically generated

Hình 3. Sơ đồ khối hàm Open

* Hàm menu:

Diagram

Description automatically generated

Hình 4. Sơ đồ khối hàm Menu

* Hàm seats:

Diagram

Description automatically generated

Hình 5. Sơ đồ khối hàm seats

* Hàm updateSeats:

Diagram

Description automatically generated

Hình 6. Sơ đồ khối hàm updateSeats

* Hàm airline

Diagram

Description automatically generated

Hình 7. Sơ đồ khối hàm airline

* Hàm check

Diagram

Description automatically generated

Hình 8. Sơ đồ khối hàm check

1. CHƯƠNG TRÌNH VÀ KẾT QUẢ
   1. Tổ chức chương trình

* Các thư viện dùng trong chương trình:
* iostream
* windows.h
* string.h
* Các hàm trong chương trình:
* Hàm in thông tin đồ án(Tên đề tài, GV hướng dẫn, Tên SV…):

*void Intro();*

* Hàm lấy thông tin sơ đồ chỗ ngồi từ file output.txt(Sơ đồ chỗ ngồi được biểu diễn bằng một mảng 2 chiều)

*void read(int output[26][10]);*

* Hàm nhập mã Opening Code mỗi lần nhân viên cần đặt chỗ cho khách trong giờ Checkin:

*void open();*

* Hàm hiển thị các chức năng chính của chương trình bao gồm xem vị trí chỗ ngồi, đặt chỗ, thoát khỏi chương trình:

*void menu();*

* Hàm hiển thị sơ đồ chỗ ngồi trên máy bay:

*void seats(char arr[26][10]);*

* Hàm cập nhật thông tin chỗ ngồi(Ở mỗi vị trí đã được đặt chỗ sẽ được kí hiệu là ‘x’)

*void updateSeat(char arr[26][10],int row,int col, string s,int ticket[]);*

* Hàm đặt chỗ:

*void airline(char arr[26][10]);*

* Hàm in thông tin vào file:

*void updateFile(int output[26][10], int row, int col, int ticket[]);*

* Hàm kết thúc chương trình:

*void shutdown();*

* 1. Ngôn ngữ cài đặt

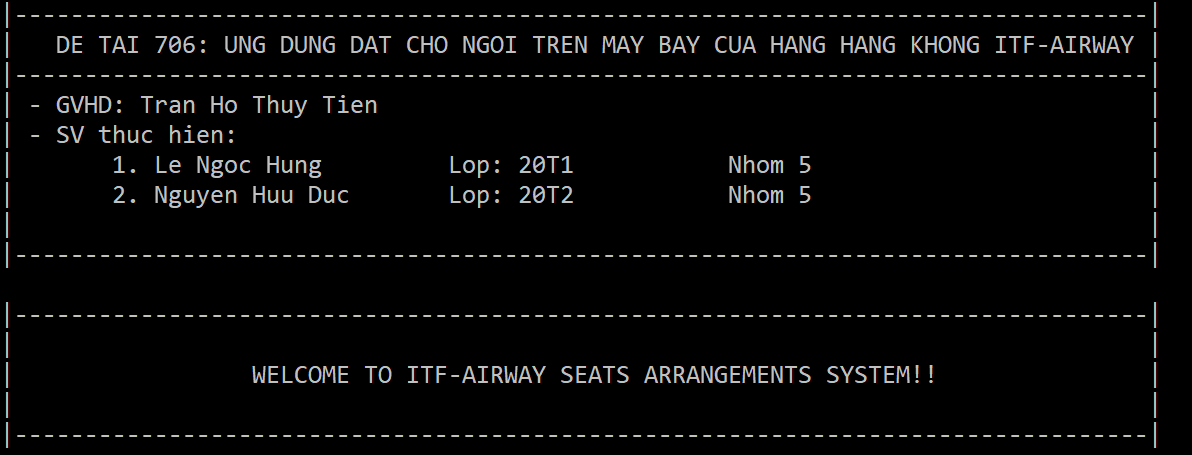
- Chương trình được cài đặt bởi ngôn ngữ C++

**4.2.1 Giới thiệu**

**-** C++ là một loại ngôn ngữ lập trình bậc trung (middle-level). Đây là ngôn ngữ lập trình đa năng được tạo ra bởi Bjarne Stroustrup như một phần mở rộng của ngôn ngữ lập trình C, hoặc "C với các lớp Class", Ngôn ngữ đã được mở rộng đáng kể theo thời gian và C ++ hiện đại có các tính năng: lập trình tổng quát, lập trình hướng đối tượng, lập trình thủ tục, ngôn ngữ đa mẫu hình tự do có kiểu tĩnh, dữ liệu trừu tượng, và lập trình đa hình, ngoài ra còn có thêm các tính năng, công cụ để thao tác với bộ nhớ cấp thấp. Từ thập niên 1990, C++ đã trở thành một trong những ngôn ngữ thương mại ưa thích và phổ biến của lập trình viên.

**4.2.2 Tổng quan**

* Vai trò đặc biệt quan trọng của bộ đôi 2 ngôn ngữ lập trình C và C++ là phục vụ cho học lập trình cơ bản. Bởi vì đây là một ngôn ngữ lập trình bậc trung. Hầu hết các trường đào tạo công nghệ thông tin ở Việt Nam đều dùng 2 ngôn ngữ này làm môn cơ sở ngành.
* Môn lập trình C giúp bạn có nền tảng với kỹ thuật lập trình, các kiến thức cơ bản và tư duy lập trình. Một lập trình C++ cung cấp kiến thức về lập trình hướng đối tượng. Cả 2 ngôn ngữ này cũng được sử dụng để học các môn học về cấu trúc dữ liệu và giải thuật.
* Nếu bạn có kiến thức tốt về 2 ngôn ngữ này, thêm với kiến thức về cấu trúc dữ liệu và giải thuật. Việc học các kiến thức lập trình mới của bạn sẽ trở nên đơn giản hơn rất nhiều. Kiến thức mới ở đây có thể là công nghệ mới, ngôn ngữ lập trình mới,…
* Ngôn ngữ C++ được Bjarne Stroustrup phát triển từ ngôn ngữ C từ cuối thập niên 1970.
* C++ là một phiên bản mở rộng của ngôn ngữ C, kết hợp tất cả các tính năng đã có của C.
* C++ được coi như là ngôn ngữ bậc trung (middle-level), kết hợp các đặc điểm và tính năng của ngôn ngữ bậc cao và bậc thấp.
* C++ có thể dùng để lập trình nhúng, lập trình hệ thống, hoặc những ứng dụng, game…
* C++ là ngôn ngữ "đa hướng". Nghĩa là nó hướng cấu trúc giống C và có thêm một tính năng cực kỳ quan trọng đó là tính năng hướng đối tượng. Các bạn sẽ được học phần hướng đối tượng của C++ trong serial Lập trình hướng đối tượng C++.
* C++ là một trong những ngôn ngữ lập trình phổ biết trên thế giới.
  1. Kết quả
     1. Giao diện chính của chương trình



Hình 9. Giao diện mở đầu chương trình

Text

Description automatically generated

Hình 10. Chương trình khi nhân viên nhập sai mã Opening Code đủ 5 lần

Text

Description automatically generated

Hình 11. Chương trình khi nhân viên nhập sai mã Flight Code

Text

Description automatically generated

Hình 12. Chương trình khi nhân viên nhập đúng mã Opening Code và mã Flight Code

* + 1. Kết quả thực thi của chương trình
* Chức năng 1: Hiển thị các vị trí đã đặt và chưa đặt trên máy bay(Vị trí đã đặt được kí hiệu là ‘x’)

Background pattern

Description automatically generated

Hình 13. Hiển thị sơ đồ chỗ ngồi

* Chức năng 2: Tiến hành đặt chỗ

Shape, arrow

Description automatically generated

Hình 14. Chương trình hiển thị sơ đồ chỗ ngồi để biết được chỗ nào còn trống

A black screen with white text

Description automatically generated with low confidence

Hình 15. Nhập sai chỗ vì đã quy ước chỗ ngồi là 2 kí tự, kí tự đầu tiên là chữ cái in hoa

Background pattern

Description automatically generated

Hình 16. Đặt chỗ thành công

* Chức năng 3: Thoát khỏi chương trình

Text

Description automatically generated

Hình 17. Thoát khỏi chương trình

Background pattern

Description automatically generated

Hình 18. Chỗ ngồi khi đã sắp đủ

Text, table

Description automatically generated

Hình 19. File ánh xạ mã vé tương ứng với chỗ ngồi đã đặt khi chuyến bay hoàn chỉnh

- Ở file output.txt, chỗ ngồi sẽ được kí hiệu theo từng hàng được đánh số từ 1 đến 26 vì mã vé của khách hàng được quy ước là số nguyên nên không thể xuất ra file kiểu vừa kí tự vừa kiểu số nguyên được.

* + 1. Nhận xét đánh giá
* Nhờ sự hướng dẫn tận tình của cô Trần Hồ Thủy Tiên, nhóm chúng em đã hoàn thành đồ án lập trình tính toán này. Qua đó, chúng em đã có cái nhìn đầy đủ, sâu sắc hơn về việc xây dựng một ứng dụng giải quyết vấn đề cụ thể, các thao tác với mảng, xử lí chuỗi, làm việc với file, cách tối ưu hóa chương trình để thân thiện với người sử dụng hơn.
* Bên cạnh đó, chúng em cũng đã phát triển thêm một số kỹ năng như làm việc nhóm, trình bày, thiết kế, giải quyết vấn đề, giao tiếp… để làm nền tảng cho việc thực hiện các đồ án lớn hơn say này.

1. KẾT LUẬN VÀ HƯỚNG PHÁT TRIỂN
   1. Kết luận

* Sau quãng thời gian tìm hiểu và thực hiện, nhóm chúng em đã cơ bản hoàn thiện chương trình và đã hoạt động đúng theo yêu cầu của đề tài, bao gồm hiển thị sơ đồ chỗ ngồi trên máy bay, đặt chỗ trong giờ Checkin chuyến bay cho hành khách.
  1. Hướng phát triển
* Khắc phục các lỗi phát sinh (nếu có)
* Xây dựng thêm nhiều chức năng như hiển thị địa điểm đi-đến, thời gian…
* Xây dựng chương trình có ứng dụng đồ họa để người sử dụng dễ tiếp cận hơn.

TÀI LIỆU THAM KHẢO

**[1] Lập trình C++:**

[***https://vi.wikipedia.org/wiki/C%2B%2B***](https://vi.wikipedia.org/wiki/C%2B%2B)

**[2] Làm việc với File:**

[***https://daynhauhoc.com/t/file-va-cac-thao-tac-co-ban-voi-file-trong-c/35276***](https://daynhauhoc.com/t/file-va-cac-thao-tac-co-ban-voi-file-trong-c/35276)

**[3] Làm việc với mảng 2 chiều:**

[***http://www.cplusplus.com/forum/beginner/9379/***](http://www.cplusplus.com/forum/beginner/9379/)

PHỤ LỤC

1. Hàm Intro

A picture containing text

Description automatically generated

Hình 20. Code hàm Intro

1. Hàm read

Text

Description automatically generated

Hình 21. Code hàm read

1. Hàm Open

Graphical user interface, text

Description automatically generated with medium confidence

Hình 22. Code hàm open

1. Hàm Menu

Timeline

Description automatically generated

Hình 23. Code hàm menu

1. Hàm seats

Text, letter

Description automatically generated

Hình 24. Code hàm seats

1. Hàm updateSeat

Text

Description automatically generated

Hình 25. Code hàm update Seats

1. Hàm airline

Text

Description automatically generated

Hình 26. Code hàm airline

1. Hàm Check

Graphical user interface, text, application, email

Description automatically generated

Hình 27. Code hàm check

1. Hàm updateFile

Text

Description automatically generated

Hình 28. Code hàm updateFile