

TRƯỜNG ĐẠI HỌC BÁCH KHOA
KHOA CÔNG NGHỆ THÔNG TIN



ĐỒ ÁN LẬP TRÌNH TÍNH TOÁN

**XÂY DỰNG ỨNG DỤNG ĐẶT MÓN VÀ
THANH TOÁN ĐƠN HÀNG**

Giáo viên hướng dẫn: GVC. ThS. Trần Hồ Thủy Tiên

Sinh viên thực hiện:

Nguyễn Thị Thu Phương
Trần Vũ Hoàng

LỚP: 23T_DT2
LỚP: 23T_B2

NHÓM: 10
NHÓM: 10

Đà Nẵng, 06/2024

MỤC LỤC

DANH MỤC HÌNH ẢNH	i
MỞ ĐẦU	ii
1 TỔNG QUAN ĐỀ TÀI.....	1
2 CƠ SỞ LÝ THUYẾT	2
2.1 Ý tưởng.....	2
2.2 Cơ sở lí thuyết	2
2.2.1 Kiểu cấu trúc	2
2.2.2 Con trỏ.....	2
2.2.3 Hàm.....	3
2.2.4 Thư viện	4
2.2.5 Thuật toán	4
3 TỔ CHỨC CẤU TRÚC DỮ LIỆU VÀ THUẬT TOÁN	5
3.1 Phát biểu bài toán	5
3.2 Cấu trúc dữ liệu	6
3.3 Thuật toán.....	6
3.3.1 Thuật toán thêm món ăn vào danh sách món.....	6
3.3.2 Thuật toán xác nhận nhân viên	7
3.3.3 Chức năng xóa món khỏi Menu.....	7
3.3.4 Chức năng in Menu ra màn hình.....	8
3.3.5 Chức năng tính toán và xuất hóa đơn	8
3.3.6 Chức năng thay đổi danh sách món ăn	8
3.3.7 Chức năng lấy thông tin món ăn từ file cho trước	8
3.3.8 Chức năng ghi dữ liệu vào file.....	9

3.3.9	Chức năng lựa chọn món ăn	9
3.3.10	Chức năng đăng xuất.....	10
3.3.11	Các chức năng phụ khác	10
4	CHƯƠNG TRÌNH VÀ KẾT QUẢ.....	11
4.1	Tổ chức chương trình	11
4.2	Xây dựng chương trình.....	12
4.3	Kết quả	13
4.3.1	Giao diện chính	13
4.3.2	Giao diện thực thi.....	15
4.3.3	Nhận xét và đánh giá.....	20
5	KẾT LUẬN VÀ HƯỚNG PHÁT TRIỂN.....	22
5.1	Kết luận	22
5.2	Hướng phát triển.....	22
	TÀI LIỆU THAM KHẢO	23
	PHỤ LỤC	24

DANH MỤC HÌNH ẢNH

Hình 2.1. Cấu trúc dữ liệu struct.....	2
Hình 2.2. Con trỏ trỏ đến struct	3
Hình 2.3. Hàm trong C.....	3
Hình 4.1. Tổ chức chương trình.....	11
Hình 4.2. Giao diện đăng nhập ứng dụng	13
Hình 4.3. Giao diện đăng nhập thành công	13
Hình 4.4. Giao diện khi nhập sai mã đăng nhập.....	13
Hình 4.5. Giao diện mở đầu ứng dụng.....	13
Hình 4.6. Giao diện đăng xuất	14
Hình 4.7. Đăng nhập thành công	14
Hình 4.8. Đăng nhập không thành công	14
Hình 4.9. Giao diện đặt món ăn	15
Hình 4.10. Lỗi sai ID món	15
Hình 4.11. Lỗi số lượng không hợp lệ.....	15
Hình 4.12 Giao diện xác nhận yêu cầu khách hàng.....	16
Hình 4.13. Giao diện thay đổi danh sách món ăn.....	16
Hình 4.14. Chức năng 1 thay đổi số lượng 1 món ăn trong danh sách.....	17
Hình 4.15. Chức năng 2 xóa 1 món ra khỏi danh sách.....	17
Hình 4.16. Chức năng 3 thêm món ăn vào danh sách	18
Hình 4.17. Giao diện hóa đơn.....	19
Hình 4.18. Giao diện lướt đặt món tiếp theo	19
Hình 4.19. File dat tổng hợp dữ liệu đơn hàng.....	20

MỞ ĐẦU

Qua 2 kì học tập tại trường, với những kiến thức đã học được qua các môn như Kỹ thuật lập trình, cấu trúc dữ liệu, chúng em đã có những hiểu biết nền tảng để xây dựng nên những ứng dụng đơn giản. Trong đồ án lập trình tính toán lần này, chúng em sẽ áp dụng những kiến thức đã học được để xây dựng hoàn thiện một ứng dụng cơ bản.

Chúng em lựa chọn xây dựng ứng dụng đặt món ăn và thanh toán đơn hàng sử dụng cho nhân viên tại quầy hàng thông qua ngôn ngữ C. Ứng dụng với giao diện đơn giản và một số chức năng cơ bản những vẫn đảm bảo giúp cho nhân viên dễ dàng đặt món. Đề tài này sẽ là nền tảng cho những bước phát triển tiếp theo để chúng em ngày ngày càng nâng cao trình độ và kỹ năng trong việc xây dựng các ứng dụng.

Đồ án này sẽ không thể hoàn thành được nếu thiếu sự hỗ trợ của các thầy cô hướng dẫn cũng như những người đã tư vấn cho chúng em trong quá trình chúng em làm đồ án. Nhóm chúng em mong rằng đây sẽ là minh chứng cho thành quả dạy dỗ của thầy cô cũng như nỗ lực học tập của chúng em trong suốt thời gian vừa qua.

Chúng em xin chân thành cảm ơn!

1 TỔNG QUAN ĐỀ TÀI

Đề bài: Xây dựng ứng dụng đặt món ăn và thanh toán đơn hàng.

Mục tiêu:

- Sử dụng các thao tác với file.
- Nhiều vấn đề cần giải quyết với các biến, vòng lặp và ra quyết định.

Yêu cầu:

- Viết chương trình bằng C/C++ để giải quyết bài toán
- Viết các hàm để xây dựng các chức năng

Mô tả chung về đề tài:

- Bắt đầu một ngày, thu ngân nhập vào một mã giao dịch.
- Chương trình in ra Menu các món ăn của nhà hàng. Cho phép khách hàng nhập vào các món ăn cần đặt với số lượng tương ứng. Chương trình sẽ tính tổng, giảm giá nếu có và in ra hóa đơn.
- Khách hàng được phép nhập tối đa 5 món và nhập vào 00 khi không chọn thêm món. Vì vậy, khi khách hàng đã chọn đủ 5 món thì tiến trình chọn món cũng kết thúc ngay lập tức và in ra Hóa đơn mà không cần khách hàng phải nhập vào 00.
- Khách hàng sẽ được giảm giá 25% nếu Tổng đơn hàng ≥ 2 triệu.
- Kết thúc một ngày thu ngân sẽ nhập vào mã giao dịch trên. Chương trình thống kê in ra toàn bộ các đơn hàng đã có trong ngày và ghi ra file.

Ngôn ngữ:

Ứng dụng được lập trình bằng ngôn ngữ C.

2 CƠ SỞ LÝ THUYẾT

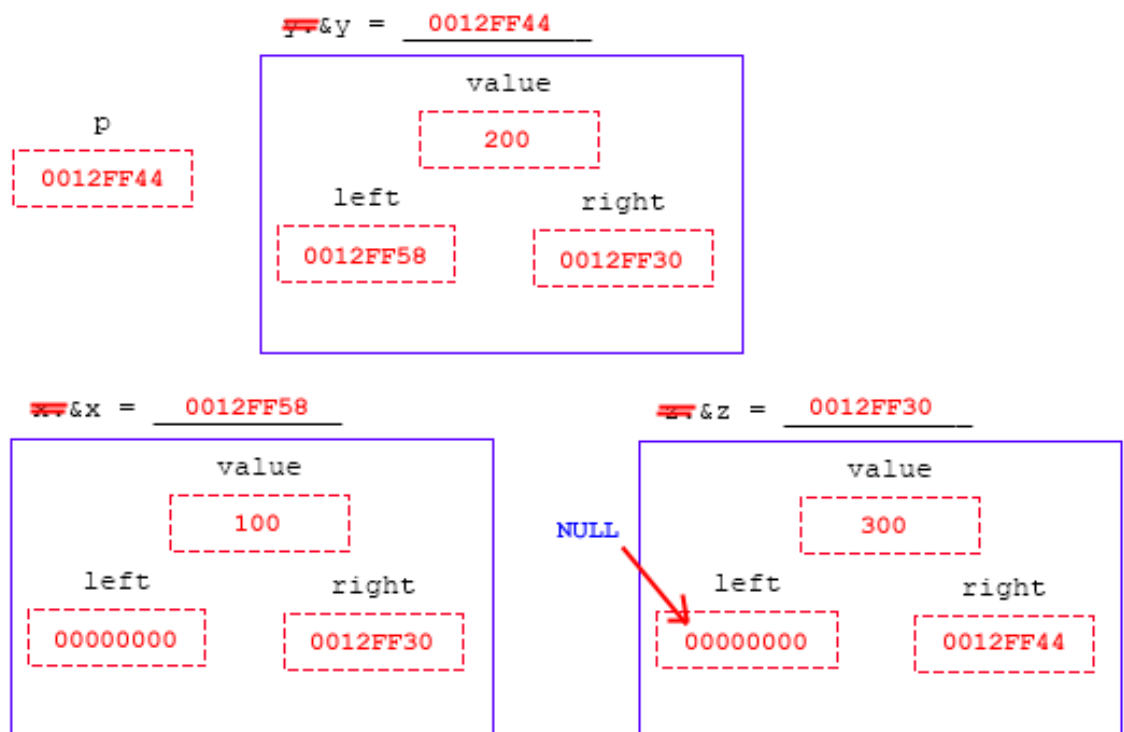
2.1 Ý tưởng

Đề tài “**Xây dựng ứng dụng đặt món và thanh toán đơn hàng**” với đối tượng hướng đến là nhân viên tại quầy. Chính vì thế, ứng dụng cần đòi hỏi sự linh hoạt, chính xác, tối ưu các thao tác hoạt động. Nhìn chung, ứng dụng sẽ có những chức năng như sau: đăng nhập và đăng xuất, đặt món ăn, xác nhận và thay đổi món ăn, tính tiền, xuất hóa đơn.

2.2 Cơ sở lý thuyết

2.2.1 Kiểu cấu trúc

Struct là một kiểu dữ liệu mà tự người dùng định nghĩa ra bằng cách gộp nhiều kiểu dữ liệu có sẵn lại nhằm mục đích có thể mô tả nhiều trường thông tin của đối tượng cần lưu. Trong struct người dùng sẽ liệt kê các trường thông tin cần lưu cho cấu trúc mà người đó đang xây dựng kèm theo kiểu dữ liệu của trường thông tin đó.



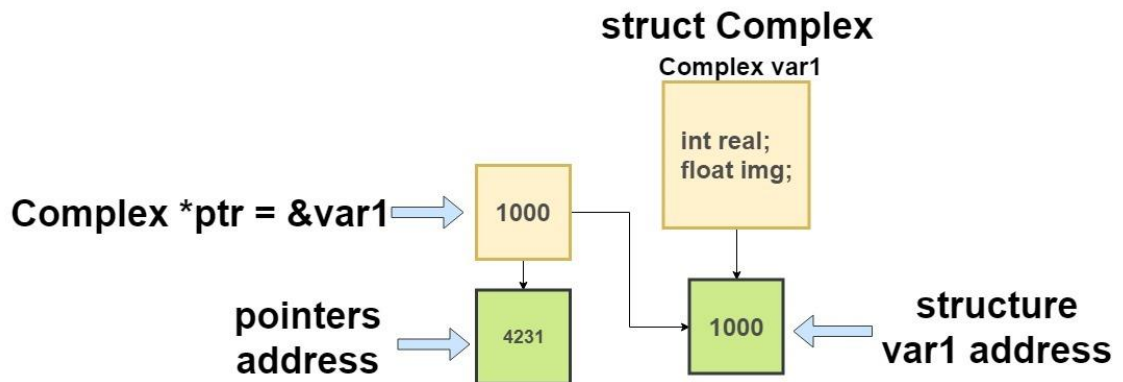
Hình 2.1. Cấu trúc dữ liệu struct

2.2.2 Con trỏ

Con trỏ là một biến trỏ đến địa chỉ của một biến khác của bất kỳ loại dữ liệu như int, char, float... Tương tự như vậy, chúng ta có thể có một con trỏ đến struct. Có hai cách để truy cập các thành phần của cấu trúc bằng cách sử dụng con trỏ:

Sử dụng toán tử * (dấu sao) và toán tử dấu "." (dấu chấm)

Sử dụng toán tử -> (mũi tên)



Hình 2.2. Con trỏ trỏ đến struct

2.2.3 Hàm

Hàm (function) là một tập hợp các xử lý nhằm thực hiện một chức năng cụ thể nào đó trong chương trình. Hàm cho phép bạn kết hợp các xử lý khác nhau thành một và đặt tên cho nó. Sau khi tạo và đặt tên cho một hàm, chúng ta chỉ cần gọi tên hàm ra mỗi khi cần sử dụng đến nó trong chương trình.

Sau khi xây dựng hàm xong để hàm có thể thực thi thì cần gọi nó trong hàm main và truyền cho nó tham số nếu cần.

```
#include <stdio.h>

void greet(){
    // code
}

int main(){
    .. .. .
    greet(); // Function call
    .. .. .
}
```

Hình 2.3. Hàm trong C

2.2.4 Thư viện

Một số thư viện có thể hỗ trợ cho chương trình:

- Thư viện `<stdio.h>`: Thư viện chứa các hàm vào/ ra chuẩn: `printf()`, `scanf()`, `getc()`, `putc()`, `gets()`, `puts()`, `fflush()`, `fopen()`, `fclose()`, `fread()`, `fwrite()`, `getchar()`, `putchar()`, `getw()`, `putw()`,...
- Thư viện `<stdlib.h>`: Thư viện cung cấp nhiều hàm tiện ích cơ bản, bao gồm các hàm quản lý bộ nhớ, chuyển đổi kiểu dữ liệu, tạo số ngẫu nhiên, và các hàm thoát chương trình. Một số chức năng chính của thư viện này: quản lý bộ nhớ, chuyển đổi kiểu dữ liệu, tạo số ngẫu nhiên, thoát chương trình,...
- Thư viện `<string.h>`: Thư viện này chứa các nguyên mẫu của hai họ hàm: Quản lý bộ nhớ và xử lý chuỗi ký tự. Một số hàm phổ biến như: `strcpy(char *dest, const char *src)`, `strstr(const char *haystack, const char *needle)`, ...
- Thư viện `<time.h>`: Thư viện này cung cấp các hàm và kiểu dữ liệu để xử lý và thao tác với thời gian và ngày tháng. Dưới đây là một số hàm phổ biến: `time(time_t *timer)`, `localtime(const time_t *timer)`,...
- Thư viện `<windows.h>`: Là một thư viện tiêu chuẩn trong lập trình Windows API, cung cấp các hàm, macro, kiểu dữ liệu và cấu trúc cần thiết để lập trình ứng dụng trên hệ điều hành Windows. Dưới đây là một số tính năng chính của thư viện: quản lý cửa sổ vào giao diện người dùng, quản lý tệp và thư mục,...

2.2.5 Thuật toán

Sử dụng thuật toán tìm kiếm linear search.

Linear search là một thuật toán tìm kiếm cơ bản dùng để tìm kiếm một phần tử trong mảng hoặc danh sách. Ý tưởng thuật toán là duyệt qua từng phần tử của mảng từ đầu đến cuối và so sánh từng phần tử với giá trị cần tìm. Nếu tìm thấy, thuật toán trả về vị trí của phần tử đó, nếu không nó trả về một giá trị chỉ ra rằng phần tử không có trong mảng.

3 TỔ CHỨC CẤU TRÚC DỮ LIỆU VÀ THUẬT TOÁN

3.1 Phát biểu bài toán

Bài toán đề tài “Xây dựng chương trình đặt món và thanh toán đơn hàng” bao gồm các chức năng sau:

- Thực hiện đặt món:
 - + Thêm món: Cho phép người dùng thêm món ăn cần đặt với số lượng tương ứng.
 - + Xóa món: Cho phép người dùng xóa món ăn không cần nữa.
 - + Thay đổi số lượng: Cho phép người dùng thay đổi số lượng 1 món trong danh sách đã chọn trước đó.
 - + Tính toán hóa đơn: Chương trình sẽ dựa vào các thông tin món ăn và số lượng để tính tổng, giảm giá nếu có.
 - + In hóa đơn: Sau khi xác nhận với khách hàng không có sự thay đổi chương trình sẽ in hóa đơn ra màn hình.
- Quản lý giao dịch: Kết thúc phiên làm việc danh sách các giao dịch trong phiên sẽ được ghi vào file.

Input:

- Thông tin mã giao dịch
- Trình yêu cầu nhập mã giao dịch và kiểm tra.
- Thông tin món ăn của quán được lưu ở file có cấu trúc: <ID>. <tên món> <giá thành>
- Yêu cầu nhập món ăn cần đặt và số lượng
- Yêu cầu thực hiện sửa đổi đơn hàng: xóa món, thêm món, thay đổi số lượng món.

Output:

- Thông tin đơn hàng sau khi đặt hàng: tên món, số lượng, giá món, tổng giá, khuyến mãi, thành tiền.
- Danh sách thông tin đặt hàng (ID món, số lượng, tổng giá, khuyến mãi, thành tiền).

3.2 Cấu trúc dữ liệu

Để xây dựng chương trình cấu trúc dữ liệu cho bài toán quản lý danh sách món bao gồm các thành phần:

‘FOOD’ là cấu trúc đại diện cho một món ăn. Bao gồm:

- ‘ID’: Mã số định danh của món ăn.
- ‘name’: tên món ăn.
- ‘price’: Giá của món ăn.
- ‘amount’: Số lượng của món ăn.
- ‘total’: Tổng giá trị của món ăn (được tính bằng giá nhân với số lượng).

‘MENU’: là cấu trúc đại diện cho một danh sách món ăn. Bao gồm:

- ‘count’: Số lượng món ăn hiện có trong thực đơn.
- ‘max_size’: Số lượng món ăn tối đa thực đơn có thể chứa.
- ‘list’: Con trỏ tới một mảng các ‘FOOD’, đại diện cho danh sách các món ăn.
- ‘discount’: Số tiền giảm giá.
- ‘subtotal’: Tổng số tiền trước khi áp dụng giảm giá.
- ‘payment’: Số tiền phải trả sau khi áp dụng giảm giá.

3.3 Thuật toán

3.3.1 Thuật toán thêm món ăn vào danh sách món

- Trong hàm *void addFoodToMenu(Menu mn, Food f, int p)*; ta áp dụng thuật toán tìm kiếm tuyến tính để duyệt tìm vị trí món ăn sẽ thêm vào.
- Hàm nhận vào các đối số là: tên danh sách món ăn sẽ được thêm vào ‘mn, thông tin món ăn ‘f’, vị trí trong danh sách món ăn sẽ được thêm vào ‘p’.

Bước 1: Kiểm tra nếu số lượng món ăn hiện tại bằng ‘max_size’ hoặc ‘p’ lớn hơn số lượng món ăn hiện tại, thì thông báo ra màn hình và kết thúc hàm.

Bước 2: Duyệt qua các món ăn trong menu:

- + Nếu món ăn đã tồn tại (‘ID’ trùng), thì:
 - Cộng số lượng mới vào món ăn đã tồn tại.
 - Cập nhật tổng giá trị của món ăn.

- Cập nhật tổng giá trị của đơn hàng ('payment').
- Kết thúc hàm.
- + Nếu món ăn chưa tồn tại, thêm món ăn với vào vị trí 'p':
 - Dời các món ăn từ vị trí 'p' trở đi về sau 1 vị trí.
 - Thêm món ăn mới vào vị trí 'p'.
 - Cập nhật số lượng món ăn và tổng giá trị của đơn hàng.

3.3.2 Thuật toán xác nhận nhân viên

- Trong hàm *void signin()*; ta áp dụng thuật toán tìm kiếm để tìm kiếm người dùng.
- Nhận vào: mã nhân viên nhập vào

Bước 1: Mở tệp user.dat.

Bước 2: Hiện thị giao diện đăng nhập.

Bước 3: Lặp lại cho đến khi người dùng nhập đúng ID:

- + Yêu cầu người dùng nhập ID.
- + Kiểm tra ID có trong danh sách nhân viên hay không.

Bước 4: In thông báo đăng nhập thành công.

Bước 5: Đóng tệp.

3.3.3 Chức năng xóa món khỏi Menu

- Trong hàm *void deleteFoodOutMenu(Menu mn, int p)*; ta áp dụng thuật toán tìm kiếm tuyến tính để duyệt tìm vị trí món ăn sẽ xóa trong danh sách.
- Hàm nhận vào 2 đối số: tên danh sách cần xóa món 'mn', vị trí món cần xóa trong danh sách 'p'.

Bước 1: Kiểm tra nếu 'p' lớn hơn số lượng món ăn hiện tại, thì in ra thông báo và kết thúc hàm.

Bước 2: Trừ tổng giá trị của món ăn cần xóa khỏi payment.

Bước 3: Dời các món ăn từ vị trí p+1 trở đi về trước một vị trí.

Bước 4: Giảm số lượng món ăn hiện tại đi 1.

3.3.4 Chức năng in Menu ra màn hình

Hàm *void printMenu(Menu mn, int option)*; thực hiện như sau:

- Nếu *option == 1*:
 - + In tiêu đề "MENU TIEM BANH NGOT".
 - + In danh sách món ăn với tên và giá tiền.
- Nếu *option == 2*:
 - In tiêu đề chi tiết đơn hàng với tên món ăn và số lượng khách hàng đặt.
- Nếu *option == 3*:
 - In tiêu đề chi tiết đơn hàng với tên món ăn, giá, số lượng, và tổng giá.

3.3.5 Chức năng tính toán và xuất hóa đơn

Hàm *void payment(Menu mn)*; gồm các bước sau:

Bước 1: In tiêu đề "Invoice" và số hóa đơn và ngày hiện tại.

Bước 2: In chi tiết đơn hàng (gọi hàm *printMenu* với *option = 3*).

Bước 3: In subtotal, discount, payment.

3.3.6 Chức năng thay đổi danh sách món ăn

Hàm *void change(Menu foodList, Menu order)*; gồm các bước sau:

Bước 1: Hiển thị menu thay đổi với các lựa chọn.

Bước 2: Dựa vào lựa chọn của nhân viên:

- + Nếu chọn thay đổi số lượng món ăn:
 - Yêu cầu người dùng nhập số thứ tự món ăn cần thay đổi và số lượng mới.
 - Cập nhật số lượng và tổng giá trị của món ăn.
- + Nếu chọn xóa món ăn:
 - Yêu cầu người dùng nhập số thứ tự món ăn cần xóa.
 - Xóa món ăn khỏi đơn hàng bằng cách gọi hàm *deleteFoodOutMenu*.
- + Nếu chọn thêm món ăn:
 - Gọi hàm *selectFood* để thêm món ăn vào đơn hàng.

3.3.7 Chức năng lấy thông tin món ăn từ file cho trước

Hàm *void readMenu(Menu list)*; gồm các bước:

Bước 1: Mở tệp menu.dat.

Bước 2: Đọc từng dòng trong tệp:

- + Phân tích dòng để lấy ID, name, và price.
- + Tạo món ăn mới (gọi hàm createFood) bằng các thông tin đã được lấy ở trên.
- + Thêm món ăn vào thực đơn (gọi hàm addFoodToMenu).

Bước 3: Đóng tệp.

3.3.8 Chức năng ghi dữ liệu vào file

Hàm *void writeToFile(Menu mn)*; gồm các bước:

Bước 1: Lấy ngày hiện tại bằng cách gọi hàm day();

Bước 2: Tạo tên tệp theo định dạng login-<ngày hiện tại>.dat.

Bước 3: Mở tệp ở chế độ thêm (append).

Bước 4: Ghi thông tin hóa đơn vào tệp:

- + Số hóa đơn.
- + Danh sách món ăn với ID và số lượng.
- + Tổng giá trị, giảm giá, và tổng giá trị phải thanh toán.

Bước 5: Đóng tệp.

Bước 6: In thông báo hoàn thành đơn hàng.

3.3.9 Chức năng lựa chọn món ăn

Hàm *void selectFood(Menu foodList, Menu order)*; gồm các bước:

Bước 1: Hiển thị thực đơn (gọi hàm printMenu với option = 1).

Bước 2: Lặp lại cho đến khi người dùng nhập "00" hoặc số lượng món ăn trong đơn hàng đạt tối đa:

- + Yêu cầu người dùng nhập ID món ăn và số lượng.
- + Khởi tạo món ăn mới (gọi hàm createFood) với các tham số:
 - o 'ID' bằng lựa chọn của người dùng.
 - o 'name' sẽ dựa vào ID để tham chiếu trong danh sách.
 - o 'price' dựa vào ID trong danh sách để tham chiếu.

- ‘amount’ số lượng được người dùng nhập vào.
- + Thêm món ăn vào đơn hàng (gọi hàm `addFoodToMenu`) với các tham số:
 - ‘order’ là tên danh sách mà món ăn được gán vào.
 - Các thông tin món ăn gồm: ‘ID’, name, price, amount đã được khởi tạo ở trên.
 - Vị trí món sẽ được lưu vào cuối danh sách.

3.3.10 Chức năng đăng xuất

Hàm `void singout()`; gồm các bước:

Bước 1: Hiển thị giao diện đăng xuất.

Bước 2: Lặp lại cho đến khi người dùng nhập đúng ID đăng xuất:

- + Yêu cầu người dùng nhập ID.
- + Kiểm tra ID với ID lúc đăng nhập.

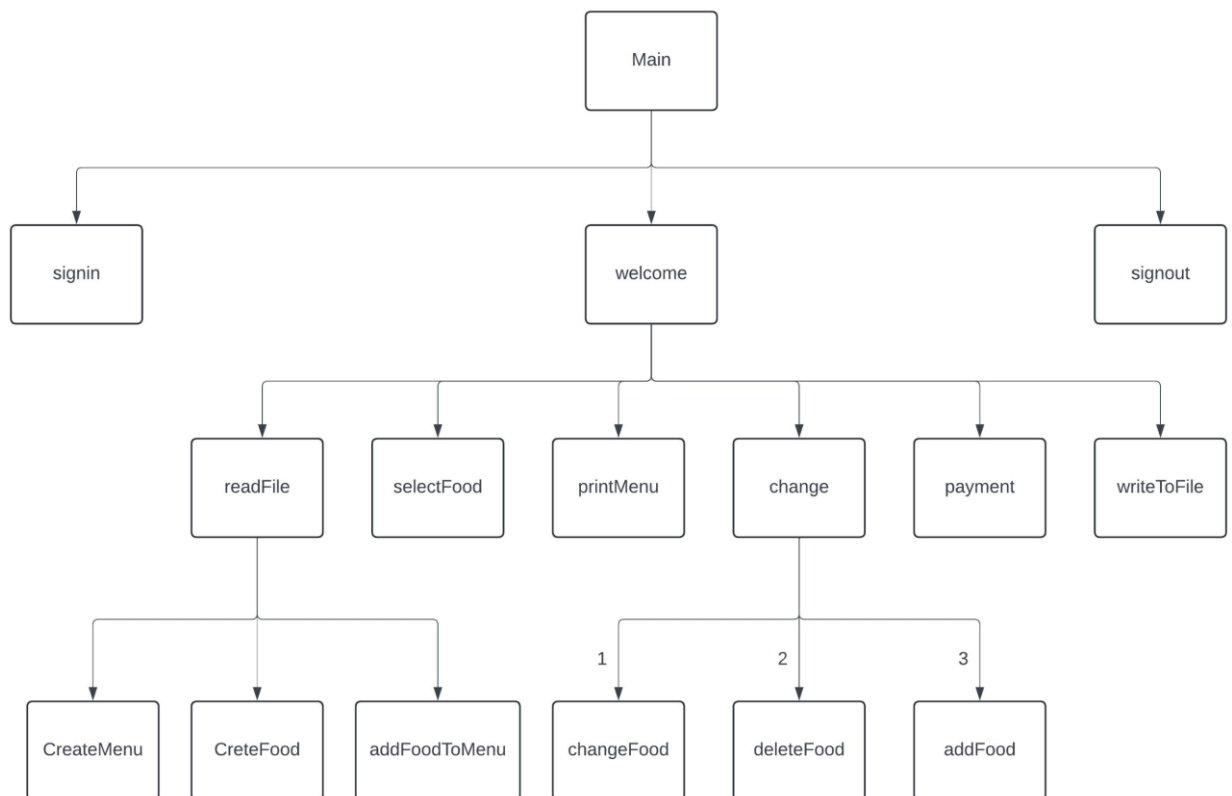
Bước 3: In thông báo đăng xuất thành công và thoát khỏi chương trình.

3.3.11 Các chức năng phụ khác

- `void today()` chức năng lấy ngày giờ định danh dd-mm-yyyy hh:mm:ss
- `void day(char *now)` chức năng lấy ngày định danh dd-mm-yyyy
- `printfWelcome()` in giao diện chào mừng đăng nhập.
- `textcolor(int x)` chức năng đổi màu chữ.

4 CHƯƠNG TRÌNH VÀ KẾT QUẢ

4.1 Tổ chức chương trình



Hình 4.1. Tổ chức chương trình

4.2 Xây dựng chương trình

Chương trình đặt món dành cho nhân viên tại quầy được viết bằng ngôn ngữ C:

```
#include <stdio.h>
#include <stdlib.h>
#include <string.h>
#include <time.h>
#include <windows.h>
#define saleOff 0.2

int orders = 0, login, logout;
struct FOOD{
    int ID;
    char name[100];
    double price;
    double amount;
    double total;
};
typedef struct FOOD *Food;

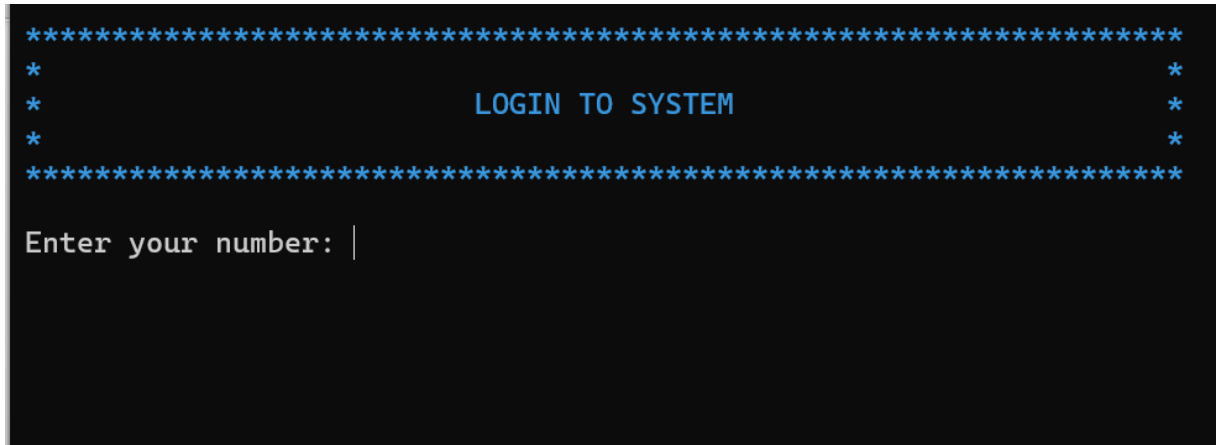
struct MENU{
    int count;
    int max_size;
    Food list;
    double discount; // Discount amount
    double subtotal; // Total before discount
    double payment; // Total after discount
};
typedef struct MENU *Menu;

Menu createMenu(int max_size); //Khởi tạo 1 danh sách món ăn
Food createFood(int ID, char name[100], double price, double amount); //Khởi tạo 1 món ăn
void addFoodToMenu(Menu mn, Food f, int p); //Thêm món ăn vào món ăn chỉ định
void deleteFoodOutMenu(Menu mn, int p); //Hàm xóa 1 món ăn ra khỏi danh sách
void printMenu(Menu mn, int option); //Hàm in danh sách món ăn
void payment(Menu mn); //Hàm tính toán hóa đơn
void change(Menu foodList, Menu order); //Hàm thay đổi danh sách món đã chọn
void readMenu(Menu list); //Hàm đọc file menu bên ngoài và gán vào danh sách
void selectFood(Menu foodList, Menu order); //Hàm chọn món
void printfwelcome();
void TextColor(int x);
void signin();
void signout();
void today();
void day(char *now);
void writeToFile(Menu mn); //Hàm lưu thông tin order vào file.
int main();
//file đầu vào
./input/user.dat //File chứa mã đăng nhập của nhân viên
./input/menu.dat //file chứa danh sách món ăn của cửa hàng
//file đầu ra
./bill/%d-%s.dat // hóa đơn của mỗi ca làm việc của một nhân viên
```

4.3 Kết quả

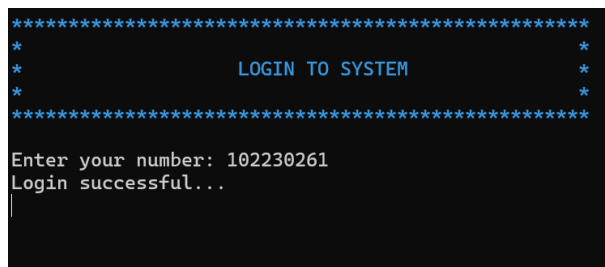
4.3.1 Giao diện chính

- Giao diện đăng nhập

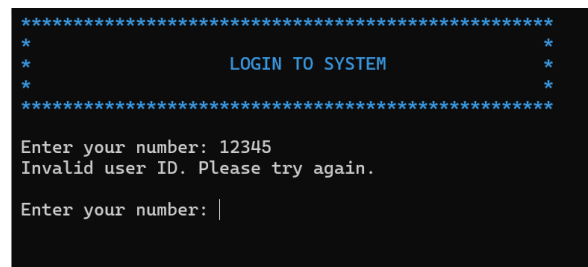


Hình 4.2. Giao diện đăng nhập ứng dụng

- Nhân viên sẽ tiến hành nhập mã nhân viên đã được cấp phát trước đó. Nhân viên chỉ đăng nhập thành công nếu nhập đúng mã, nếu sai, chương trình sẽ thông báo lỗi và yêu cầu đăng nhập lại.

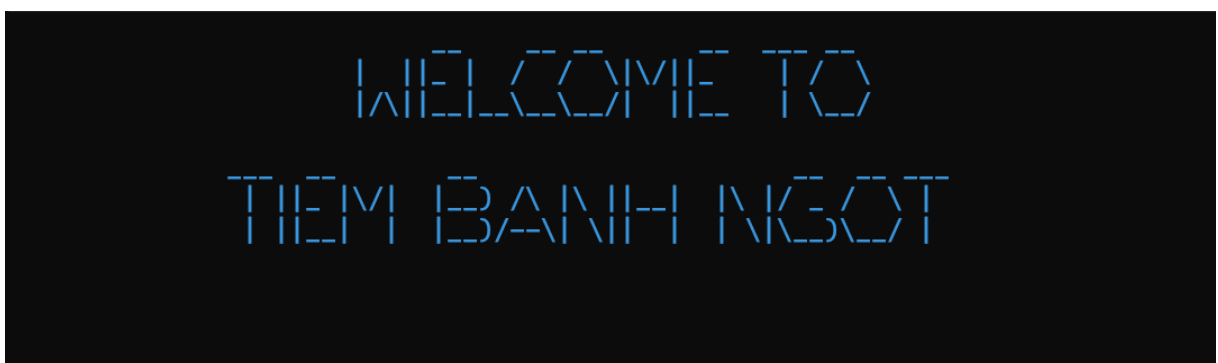


Hình 4.3. Giao diện đăng nhập thành công



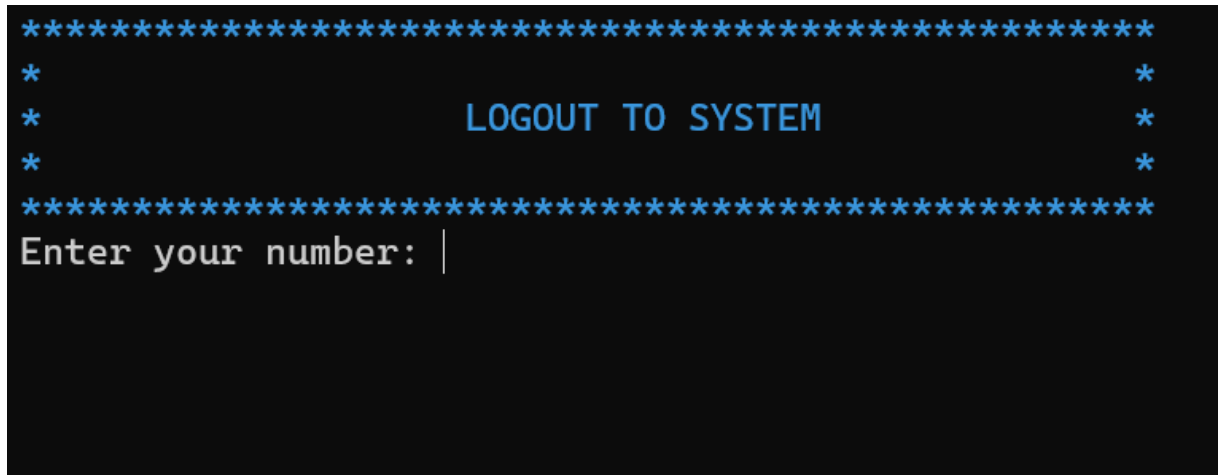
Hình 4.4. Giao diện khi nhập sai mã đăng nhập

- Giao diện mở đầu ứng dụng



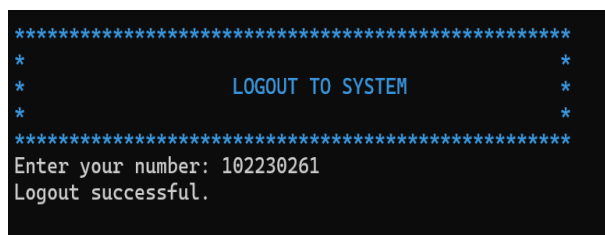
Hình 4.5. Giao diện mở đầu ứng dụng

- Sau khi đăng nhập thành công, chương trình sẽ chuyển qua giao diện mở đầu sau đó tự động chuyển qua giao diện đặt món ăn.
 - Giao diện đăng xuất

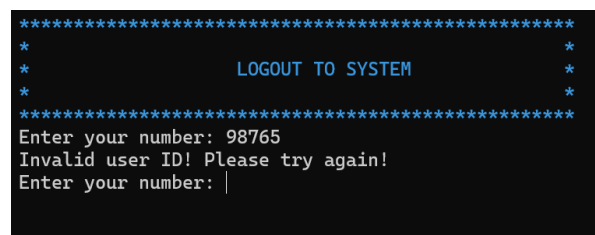


Hình 4.6. Giao diện đăng xuất

- Để kết thúc ca làm của mình, nhân viên sẽ tiến hành đăng xuất. Nhân viên phải nhập đúng mã nhân viên đã nhập ở phần đăng nhập. Nếu sai chương trình sẽ báo lỗi và hiện thị yêu cầu nhập lại.



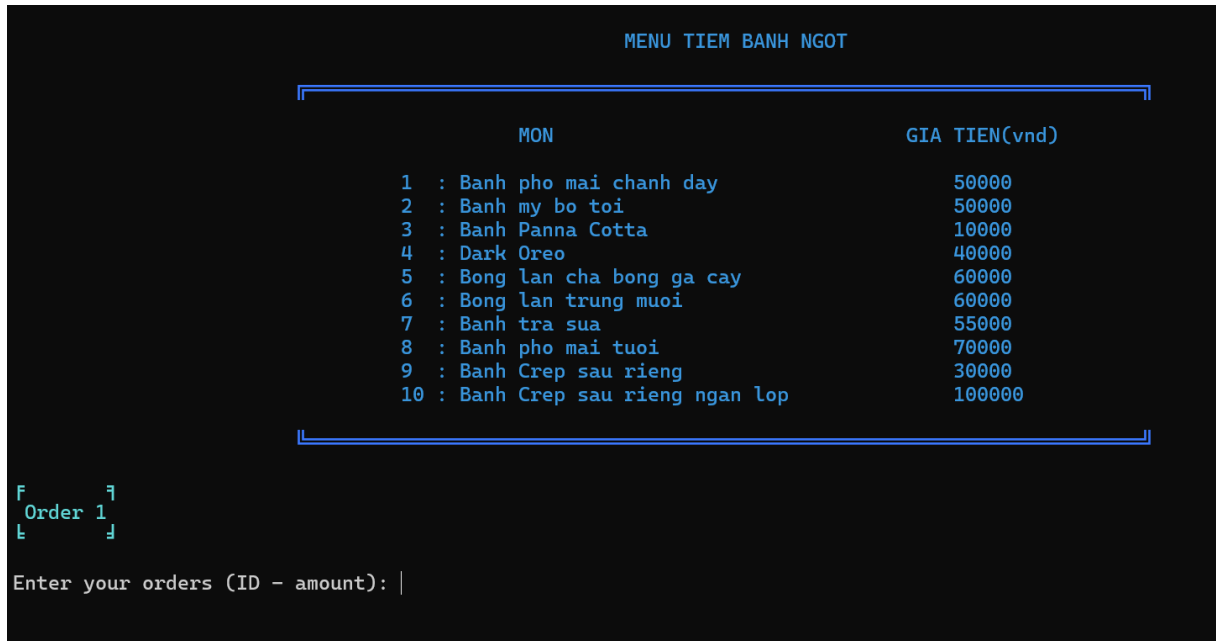
Hình 4.7. Đăng nhập thành công



Hình 4.8. Đăng nhập không thành công

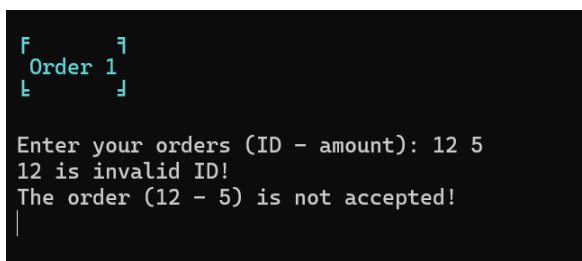
4.3.2 Giao diện thực thi

a. Giao diện đặt món ăn

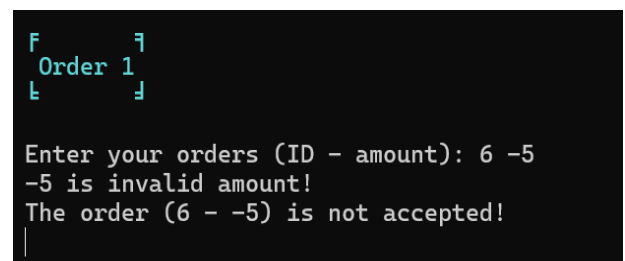


Hình 4.9. Giao diện đặt món ăn

- Sau khi truy cập thành vào chương trình, màn hình sẽ hiển thị danh sách các món ăn của quán, lượt order và yêu cầu người dùng nhập số thứ tự món và số lượng.
- Người dùng chỉ có thể nhập vào những món được hiển thị trên màn hình, với những món nằm ngoài phạm vi hoặc số lượng không hợp lệ chương trình sẽ báo lỗi.
- Người dùng tối đa chỉ có thể nhập vào các cặp số để hệ thống tự động chọn món với số lượng kèm theo và lưu vào danh sách đặt món. Mỗi lượt order chỉ được đặt tối đa 5 món.



Hình 4.10. Lỗi sai ID món



Hình 4.11. Lỗi số lượng không hợp lệ

b. Giao diện xác nhận yêu cầu khách hàng

STT	FOOD NAME	Quantity
1	Banh pho mai chanh day	2
2	Banh Panna Cotta	4
3	Bong lan cha bong ga cay	6
4	Banh tra sua	8
Do you want to change anything? (y/n)?		

Hình 4.12 Giao diện xác nhận yêu cầu khách hàng

- Khi số lượng món đạt tối đa hoặc người dùng nhập vào ‘00’ chương trình sẽ in ra danh sách các món đã chọn và hiển thị yêu cầu xác nhận lại với khách hàng.
 - Nếu chọn “y”: Thay đổi, chương trình chuyển đến thực hiện thay đổi.
 - Nếu chọn “n”: Không thay đổi, chương trình chuyển đến in hóa đơn.

c. Giao diện thay đổi món ăn theo yêu cầu khách hàng

STT	FOOD NAME	Quantity
1	Banh pho mai chanh day	2
2	Banh Panna Cotta	4
3	Bong lan cha bong ga cay	6
4	Banh tra sua	8
Do you want to change anything? (y/n)? y		
<div style="border: 1px solid black; padding: 5px;"> 1. Change the amount. 2. Delete the food 3. Add dishes. </div>		
Your Select:		

Hình 4.13. Giao diện thay đổi danh sách món ăn

- Sau khi nhân viên nhập “y” xác định thay đổi yêu cầu, chương trình sẽ hiển thị 3 lựa chọn
 - + 1: Thay đổi số lượng của 1 món.
 - + 2: Xóa 1 món.
 - + 3: Thêm món ăn.
- Người dùng sẽ chọn 1 trong 3 lựa chọn trên, nếu nhập ký tự khác thì chương trình sẽ báo lỗi vào tiếp tục chờ để nhận lệnh.

```

STT|          FOOD NAME          |          Quantity
1 |    Banh pho mai chanh day    |             2
2 |    Banh Panna Cotta          |             4
3 |    Bong lan cha bong ga cay  |             6
4 |    Banh tra sua              |             8
Do you want to change anything? (y/n)? y

┌───────────────────────────────────┐
│ 1. Change the amount.             │
│ 2. Delete the food                │
│ 3. Add dishes.                    │
└───────────────────────────────────┘

Your Select:1
Enter the dish you want to change (ID - amount): 1 4

Dish changed successfully.

Loading...|

```

Hình 4.14. Chức năng 1 thay đổi số lượng 1 món ăn trong danh sách

- Nếu lựa chọn 1, chương trình sẽ hiện thông báo yêu cầu người dùng nhập vào mã món ăn cần thay đổi và số lượng thay đổi. Nếu cặp giá trị đó hợp lệ thì chương trình sẽ thực hiện thay đổi số lượng. Ngược lại, sẽ thông báo lỗi và quay trở lại xác nhận lại với khách hàng có muốn thay đổi danh sách các món đã đặt.

```

STT|          FOOD NAME          |          Quantity
1 |    Banh pho mai chanh day    |             4
2 |    Banh Panna Cotta          |             4
3 |    Bong lan cha bong ga cay  |             6
4 |    Banh tra sua              |             8
Do you want to change anything? (y/n)? y

┌───────────────────────────────────┐
│ 1. Change the amount.             │
│ 2. Delete the food                │
│ 3. Add dishes.                    │
└───────────────────────────────────┘

Your Select:2
Enter the dish you want to cancel: 4

Dish canceled successfully.

Loading...|

```

Hình 4.15. Chức năng 2 xóa 1 món ra khỏi danh sách

- Nếu lựa chọn 2 xóa món. Chương trình sẽ hiển thị yêu cầu nhân viên nhập vào mã món cần xóa. Nếu món cần tồn tại trong danh sách thì chương trình sẽ xóa món ra khỏi danh sách đặt hàng. Ngược lại nếu món nhập vào không có trong danh sách chương trình sẽ thông báo lỗi và quay trở lại xác nhận lại với khách hàng có muốn thay đổi danh sách các món đã đặt.

```

Your Select:3

                                MENU TIEM BANH NGOT

┌────────────────────────────────────────────────────────────────────────────────┐
│                                     MON                                     GIA TIEN(vnd)│
├────────────────────────────────────────────────────────────────────────────────┤
│ 1 : Banh pho mai chanh day                                           50000    │
│ 2 : Banh my bo toi                                                    50000    │
│ 3 : Banh Panna Cotta                                                  10000    │
│ 4 : Dark Oreo                                                          40000    │
│ 5 : Bong lan cha bong ga cay                                         60000    │
│ 6 : Bong lan trung muoi                                               60000    │
│ 7 : Banh tra sua                                                       55000    │
│ 8 : Banh pho mai tuoi                                                 70000    │
│ 9 : Banh Crep sau rieng                                               30000    │
│ 10 : Banh Crep sau rieng ngan lop                                     100000   │
└────────────────────────────────────────────────────────────────────────────────┘

F      ↵
Order 1
└      ↵

Enter your orders (ID - amount):
8 7
00

Dishes added successfully.

Loading...|

```

Hình 4.16. Chức năng 3 thêm món ăn vào danh sách

- Nếu lựa chọn 3 thêm món ăn. Chương trình sẽ kiểm tra xem số lượng món ăn trong danh sách đã đạt giới hạn chưa. Nếu chưa chương trình sẽ hiển thị danh sách món của quán và yêu cầu người dùng nhập mã món và số lượng món muốn thêm vào. Số lượng món có thể thêm vào tùy thuộc vào số lượng món đã có trong danh sách đặt hàng đã đạt giới hạn chưa. Sau khi hoàn tất chương trình lại trở lại màn xác nhận yêu cầu của khách hàng.

d. Giao diện in hóa đơn

Invoice

Invoice No. 1

Date: 12-06-2024 22:45:02

FOOD NAME	Price	Quantity	Total
Banh pho mai chanh day	50000	4	200000.0
Banh Panna Cotta	10000	4	40000.0
Bong lan cha bong ga cay	60000	6	360000.0
Banh pho mai tuoi	70000	7	490000.0

Subtotal: 1090000 VND

Discount: 0 VND

Payment : 1090000 VND

Order has been completed.

Press 00 to logout

Enter any number to continue

Hình 4.17. Giao diện hóa đơn

- Khi khách hàng xác nhận không có gì thay đổi. Chương trình sẽ in thông tin chi tiết đơn hàng ra màn hình. Sau khi in hóa đơn ra màn hình. Thông tin đặt món sẽ được lưu vào file bill.dat. Chương trình sẽ đưa ra 2 lựa chọn cho nhân viên thao tác:
 - + Lựa chọn 1: Nhân viên nhấn ‘00’ để vào thoát khỏi chương trình.
 - + Lựa chọn 2: Nhập vào ký tự bất kỳ tiếp tục lượt đặt món tiếp theo.
- Khi nhân viên nhập đúng mã nhân viên đã nhập ở phần đăng nhập, thì tiến hành đăng xuất. Nếu sai chương trình sẽ báo lỗi và hiện thị yêu cầu nhập lại.

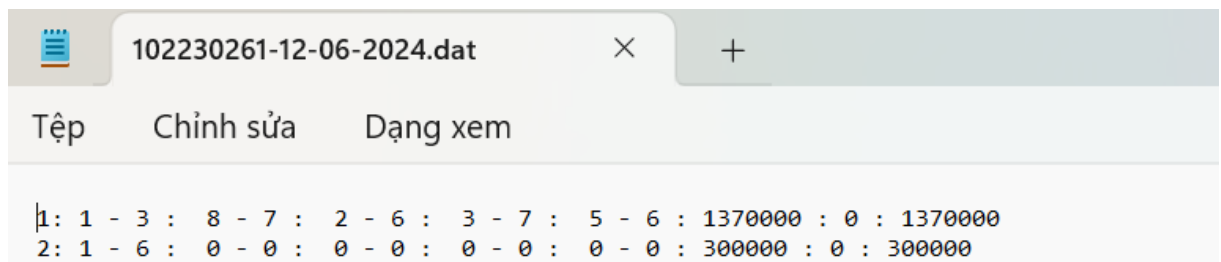
e. Giao diện thực hiện lượt đặt món tiếp theo

<

Hình 4.18. Giao diện lượt đặt món tiếp theo

Sau khi kết thúc một lượt đặt món, nếu không lựa chọn đăng xuất, chương trình sẽ chuyển qua lượt đặt món tiếp theo tạo thành 1 chu trình mới.

f. File tổng hợp dữ liệu đơn hàng



Tệp	Chỉnh sửa	Dạng xem
1: 1 - 3 : 8 - 7 : 2 - 6 : 3 - 7 : 5 - 6 : 1370000 : 0 : 1370000		
2: 1 - 6 : 0 - 0 : 0 - 0 : 0 - 0 : 0 - 0 : 300000 : 0 : 300000		

Hình 4.19. File dat tổng hợp dữ liệu đơn hàng

Tất cả những thông tin đơn hàng trong 1 ca làm việc của 1 nhân viên sẽ được lưu trong 1 file riêng.

Phần in ra file được định dạng theo cấu trúc:

<hạng order>: <ID món 1> - <số lượng> : ... : <ID món 5> - <số lượng>: <tổng>:<giảm giá>:<thành tiền>

4.3.3 Nhận xét và đánh giá

Ưu điểm

- Chức năng đầy đủ: Chương trình có các chức năng cơ bản như tạo menu, thêm món ăn vào menu, xóa món ăn khỏi menu, tính toán tổng tiền thanh toán và áp dụng giảm giá.
- Quản lý người dùng: Chương trình hỗ trợ chức năng đăng nhập và đăng xuất, giúp quản lý người dùng dễ dàng.
- Lưu trữ thông tin đơn hàng: Chương trình lưu trữ thông tin đơn hàng vào file, giúp theo dõi và quản lý đơn hàng của khách hàng.
- Giao diện người dùng rõ ràng: Chương trình sử dụng giao diện console với các thông báo và định dạng rõ ràng, giúp người dùng dễ sử dụng.
- Sử dụng cấu trúc dữ liệu hợp lý: Chương trình sử dụng cấu trúc dữ liệu struct để quản lý thông tin món ăn và menu, giúp tổ chức dữ liệu một cách có hệ thống.

Nhược điểm

- Sử dụng lệnh goto: Sử dụng lệnh goto để điều hướng trong chương trình là một thực hành không tốt, gây khó khăn trong việc theo dõi và bảo trì mã nguồn.
- Phụ thuộc vào hệ điều hành Windows: Chương trình sử dụng các hàm như sleep và TextColor, phụ thuộc vào thư viện windows.h, làm cho chương trình khó tương thích trên các hệ điều hành khác.

- Thiếu bảo mật: Chương trình không có bất kỳ biện pháp bảo mật nào để bảo vệ thông tin đăng nhập của người dùng.
- Không có giao diện đồ họa: Chương trình chỉ sử dụng giao diện dòng lệnh, hạn chế trải nghiệm người dùng so với các ứng dụng có giao diện đồ họa.

5 KẾT LUẬN VÀ HƯỚNG PHÁT TRIỂN

5.1 Kết luận

Nhìn chung, chương trình này có thể giúp cho nhân viên thao tác chọn món cho khách hàng 1 cách nhanh chóng, dễ dàng và thuận tiện. Tuy nhiên, chương trình vẫn còn tương đối sơ sài, vẫn còn nhiều tính năng cần phát triển, giao diện người dùng không bắt mắt.

5.2 Hướng phát triển

Đồ án đang dừng lại ở mức độ cơ bản, dự định hướng phát triển tiếp theo là bổ sung mã hóa đơn cho từng đơn đặt hàng, xây dựng thêm tính năng khi đăng nhập vào chương trình bằng tài khoản admin có thêm 2 tính năng là thêm nhân viên và thêm món ăn. Tiếp theo trong tài khoản admin có thể mở danh sách đặt món đã lưu vào thư mục bill, tổng kết doanh thu trong ngày, tra cứu được lịch sử đặt hàng theo mã hóa đơn nhập vào, ... Phát triển giao diện đồ họa cho chương trình để sử dụng cũng như thao tác được thuận tiện và rõ ràng hơn.

TÀI LIỆU THAM KHẢO

- [1] 28Tech Blog. [C]. “Struct - Kiểu Cấu Trúc”, từ: <https://blog.28tech.com.vn/c-struct-kieu-cau-truc>
- [2] Chu Minh Nam. “Con trỏ thao tác với struct trong C”, từ: <https://laptrinh tudau.com/con-tro-thao-tac-voi-struct-trong-c/>
- [3] Hữu Thuận. “Viết hàm thay đổi màu chữ trong màn hình console C/C++ – textColor()”, từ: <https://tuicocach.com/viet-ham-thay-doi-mau-chu-trong-man-hinh-console-c-c/>
- [4] Trần Hồ Thủy Tiên, Bài giảng Kỹ thuật lập trình.
- [5] Linear Search Algorithm - Data Structure and Algorithms Tutorials - GeeksforGeeks, từ: <https://www.geeksforgeeks.org/linear-search/?ref=outind>

PHỤ LỤC