



UNIVERSIDADE FEDERAL RURAL DO SEMI-ÁRIDO - UFERSA

BACHARELADO EM TECNOLOGIA DA INFORMAÇÃO - BTI

DISCIPLINA: ALGORITMOS E ESTRUTURAS DE DADOS I

DOCENTE: GEORGE FELIPE FERNANDES VIEIRA

DISCENTES: IZADORA LOUYZA SILVA FIGUEIREDO, LÍVIAN MARIA LUCENA

GOMES PINHEIRO, MARIA VITORIA FERNANDES ROCHA, RENATO VITOR

JUVÊNCIO LEITE, VICTOR HUGO DE OLIVEIRA

## **Manual de Uso do Sistema de Gerenciamento de Pedidos de Restaurante**

### **1. Introdução**

Este manual tem como objetivo fornecer instruções claras e objetivas para a utilização do Sistema de Gerenciamento de Pedidos para Restaurante, desenvolvido em linguagem C como parte da disciplina de Algoritmo e Estrutura de Dados I. O sistema simula o funcionamento de um restaurante, permitindo o controle do cardápio e dos pedidos. Com ele, é possível cadastrar, atualizar e remover itens do cardápio, além de criar, modificar e finalizar pedidos, acompanhando seu status em tempo real, desde "Pendente" até "Entregue".

Projetado para integrar conceitos fundamentais de programação, o sistema faz uso de ponteiros, manipulação de strings, alocação dinâmica de memória, vetores dinâmicos, structs (registros) e enumerações. Sua interface é baseada em um menu interativo, permitindo que o usuário realize operações de forma intuitiva, com validações que evitam erros de entrada e garantem um funcionamento eficiente.

### **2. Utilização do sistema**

A seguir, abordaremos as funcionalidades do sistema e as melhores práticas para aproveitá-lo ao máximo, garantindo uma experiência eficiente e sem

complicações.

## 2.1. Compilação e execução do sistema

Para compilar e executar o sistema, siga os seguintes passos:

- Primeiramente, certifique-se de ter um ambiente que suporte a linguagem C e um compilador (por exemplo, GCC). Recomenda-se o uso do **Visual Studio Code** para isso.

- Abra o **Visual Studio Code** e clone o repositório no terminal:

```
https://github.com/whorshe/SistemaGerenciamentoPedidosRestaurante
```

- Acesse a página do projeto e a pasta que contém os arquivos:

```
cd gerenciamento-restaurante
```

```
cd cardapio_pedidos
```

- Compile o programa:

```
gcc -o restaurante main.c cardapio.c pedidos.c utilitarios.c
```

- Execute o programa:

```
./restaurante
```

## 2.2. Menu Principal e Funcionalidades

Ao executar o programa, o usuário terá acesso ao menu principal, que oferece as seguintes opções:

### 1. Cadastrar Item

Permite cadastrar um novo item no cardápio. O usuário deve informar:

- Nome: Nome do item (ex.: "Salada Caesar");
- Descrição: Descrição do item (ex.: "Salada com alface, croutons e molho Caesar");

- Preço: Valor do item em reais (ex.: 25,90);
- Categoria: Escolha entre Entrada, Principal, Sobremesa ou Bebida.

## **2. Exibir Cardápio:**

Exibe todos os itens cadastrados no cardápio, com detalhes como o código do item, nome, descrição, preço e categoria.

## **3. Atualizar Cardápio:**

Permite atualizar as informações de um item já cadastrado. O usuário deve informar o código do item e fornecer os novos dados.

## **4. Remover Item do Cardápio:**

Remove um item do cardápio com base no código informado. O sistema reorganiza a lista de itens automaticamente.

## **5. Criar Pedido:**

Permite criar um novo pedido. O usuário deve informar:

- Nome do Cliente: Nome do cliente que realizou o pedido;
- Quantidade de Itens: Número de itens que o cliente deseja pedir;
- Itens do Pedido: Códigos dos itens do cardápio que serão incluídos no pedido.

O status inicial do pedido será definido como Pendente.

## **6. Gerenciar Pedidos:**

Exibe todos os pedidos realizados, com detalhes como:

- Código do Pedido: Identificador único do pedido;
- Status: Status atual do pedido (Pendente, Em Preparo, Pronto ou Entregue);

- Nome do Cliente: Nome do cliente que realizou a solicitação;
- Itens do Pedido: Lista de itens solicitados, com nome e preço de cada um;
- Valor Total: Soma dos preços dos itens do pedido.

#### **7. Alterar Status do Pedido:**

Permite alterar o status de um pedido. O usuário deve informar o código do pedido e escolher o novo status.

#### **8. Alterar Pedido:**

Permite modificar os itens de um pedido já existente. O usuário deve informar o código do pedido e os novos itens desejados.

#### **9. Remover Pedido:**

Remove um pedido com base no código informado. O sistema reorganiza a lista de pedidos automaticamente.

#### **10. Sair:**

Encerra a execução do sistema após uma confirmação binária, na qual 1 significa “sim” e 0 significa “não”.

### **2.3. Interação com o Sistema**

Todas as entradas são validadas para evitar erros. Por exemplo:

- Códigos de itens ou pedidos devem ser números válidos e existentes;
- Os preços não podem ser negativos;
- Quantidades de itens devem ser números positivos.

O sistema utiliza alocação dinâmica de memória, garantindo que os dados sejam armazenados e gerenciados de forma eficiente, mesmo com um grande número de itens ou pedidos.

## **2.4. Exemplo de Fluxo de Uso**

1. Cadastre alguns itens no cardápio (opção 1).
2. Crie um pedido para um cliente, selecionando os itens desejados (opção 5).
3. Acompanhe o status do pedido e altere-o conforme necessário (opção 6 e 7, respectivamente).
4. Exiba o cardápio ou a lista de pedidos para verificar as informações (opções 2 e 6).
5. Ao finalizar, saia do sistema (opção 10).

## **3. Considerações finais**

O Sistema de Gerenciamento de Pedidos para Restaurante foi projetado para consolidar os conceitos de programação em C, utilizando estruturas de dados e recursos como ponteiros, structs (registros), enums (enumeração) e alocação dinâmica de memória. A proposta é criar uma solução prática para gerenciar cardápios e pedidos de forma eficiente e dinâmica, permitindo a inclusão, atualização e remoção de dados de maneira fácil e intuitiva.

Dessa forma, ele se destaca pela sua interface simples e interativa, que facilita a navegação e oferece uma experiência mais agradável para o usuário. Além disso, possui validação de entradas para evitar erros durante o uso, alocação dinâmica de memória para otimizar o armazenamento e um código modularizado que facilita a manutenção e futuras expansões.

Embora o sistema já seja funcional, há espaço para melhorias, como a implementação de persistência de dados para salvar informações de cardápios e pedidos, a inclusão de relatórios de vendas e controle de estoque, e o aprimoramento de uma interface gráfica, o que poderia facilitar ainda mais a gestão dos pedidos. No fim, foi gratificante construir o código e analisar cada etapa do processo, pois permitiu aplicar e consolidar conceitos importantes de programação, além de ver a evolução do sistema de forma prática e funcional.