

Trabalho de Estrutura de Dados

**Joice da Rocha Silveira, Lorena Dias Freitas, Manuela de Oliveira Figueira,
Rafaela Canguçu Souza, Samuel Nascimento Santos**

Centro Universitário de Excelência (Unex) - Sistemas de Informação - Vitória da Conquista, BA – Brasil

Resumo. *Este trabalho mostra o desenvolvimento de um sistema de atendimento para restaurante feito em Python, rodando no console. O sistema usa listas e filas para organizar o cardápio, registrar pedidos e acompanhar seu andamento. O cardápio foi criado em listas, permitindo incluir, consultar e atualizar pratos. Já os pedidos seguem o modelo FIFO, em que o primeiro a entrar é o primeiro a ser atendido. Dessa forma, é possível acompanhar todo o ciclo de um pedido, desde a escolha dos pratos até a entrega, aplicando conceitos básicos de estruturas de dados em uma situação prática.*

Abstract. *This work presents the development of a restaurant service system built in Python, running in the console. The system uses lists and queues to manage the menu, register orders, and track their progress. The menu is implemented with lists, allowing dishes to be added, viewed, and updated. Orders follow the FIFO model, where the first one in is the first one out. In this way, the system tracks the whole order cycle, from selecting dishes to delivery, applying basic data structure concepts in a practical situation.*

1. Introdução

Estrutura de dados é a forma com que as informações são organizadas. Essa organização ajuda no processamento mais fluido por parte das máquinas, e na interpretação das informações do lado humano. Além disso, a estrutura leva em consideração muitos aspectos, considerando desde a relação dos dados até a eficiência de busca projetada no projeto. De maneira direta, podemos dizer que a estrutura de dados visa trazer mais eficiência para a implementação de um determinado dado. Ela é essencial para que as máquinas possam fazer a leitura de uma abundante quantidade de informações. Isso ajuda, especialmente, no desenvolvimento de softwares, pois, em muitos casos, eles são extensos.

2. Fundamentação Teórica

Lista: Uma lista é uma estrutura que armazena elementos de forma organizada e em sequência, permitindo que cada item seja identificado e acessado pelo seu índice. Funciona como uma coleção numerada, onde a posição de cada elemento é fixa e conhecida. As listas preservam a ordem de inserção e podem conter valores iguais, além de possibilitar alterações em qualquer ponto da sequência. Outra característica importante é a flexibilidade, já que é possível incluir ou remover itens conforme necessário. Entre as utilizações mais comuns estão: controle de tarefas em aplicações de

listas de afazeres, armazenamento de registros temporários como histórico de navegação e manutenção de dados em sistemas que exigem preservação da ordem dos elementos.

Fila: Uma fila é uma estrutura de dados que organiza elementos de forma sequencial, seguindo a lógica **FIFO (First In, First Out)**, ou seja, o primeiro elemento a entrar é também o primeiro a sair. Nessa estrutura, as inserções acontecem sempre no final, enquanto as remoções ocorrem no início, garantindo que a ordem de chegada seja respeitada. Diferente das listas, em que é possível acessar qualquer posição, na fila as operações são restritas às extremidades, tornando seu uso mais específico. As filas podem crescer ou reduzir dinamicamente conforme novos itens são adicionados ou removidos. Entre suas principais aplicações estão os sistemas de atendimento, como filas em bancos ou call centers, o gerenciamento de processos em sistemas operacionais e a organização de tarefas em ordem de execução.

Similaridade e diferenças: Listas e filas são estruturas de dados ordenadas que permitem armazenar diferentes tipos de elementos e se ajustam dinamicamente em tamanho. A principal diferença está no modo de acesso: listas permitem inserções, remoções e consultas em qualquer posição, enquanto filas seguem a lógica FIFO (First In, First Out), em que os elementos entram no final e saem pelo início, garantindo o processamento na ordem de chegada.

3. Metodologia

O sistema foi desenvolvido em Python como uma aplicação de terminal interativa, voltada para o gerenciamento de pedidos e produtos de maneira prática e eficiente. O objetivo principal foi criar uma solução funcional, modular e de fácil uso, em que menus orientam o usuário na execução de operações como cadastro, atualização e consulta de itens, criação de pedidos e acompanhamento do status dos mesmos.

O desenvolvimento foi estruturado em cinco partes principais, garantindo modularidade e separação clara entre o controle de produtos e o gerenciamento de pedidos. O menu de itens permite adicionar, atualizar e visualizar produtos disponíveis, enquanto o menu de pedidos possibilita criar novos pedidos, adicionar itens ao carrinho, aplicar descontos e alterar o status conforme o andamento.

Para armazenamento e manipulação dos dados, o sistema utiliza listas, permitindo registrar informações de forma dinâmica e acessível. A lógica de navegação e operação foi construída com estruturas condicionais, criando um fluxo simples e eficiente, assegurando que cada ação seja validada antes de prosseguir. Toda a interação ocorre via terminal, simulando um ambiente de uso real para pequenos comércios ou serviços de delivery, onde o controle rápido e organizado de produtos e pedidos é essencial.

4. Resultados e Discussões

4.1 Principais resultados:

- **Cadastro e Consulta de Itens:** Permite registrar e visualizar itens com nome, descrição, preço e estoque, validando corretamente entradas numéricas.

- **Atualização de Itens:** Possibilita alterar atributos de itens cadastrados, mantendo valores anteriores quando desejado.
- **Criação e Processamento de Pedidos:** Permite adicionar múltiplos itens, calcular total com desconto e gerenciar pedidos pendentes, aceitos ou rejeitados.
- **Atualização de Status e Consulta de Pedidos:** Permite mover pedidos entre os status ACEITO → PRONTO → ENTREGUE, garantindo acompanhamento claro de cada pedido.

4.2 Discussão:

O uso de listas e tuplas mostrou-se eficiente para armazenamento e manipulação dos dados. O sistema, embora simples e baseado em terminal, cumpre os objetivos propostos, demonstrando controle de cadastro, filas de pedidos e atualização de status. Limitações incluem ausência de persistência de dados e interface gráfica.

4.3 Prints da Execução no Console

A seguir, alguns exemplos da execução real do programa no terminal:

```
1 - Cadastrar item
2 - Consultar itens
0 - Sair
Escolha: 1

Cadastrar novo item do cardápio
Nome do item: Pão de forma
Descrição: integral
Preço: R$ 6,50
Digite um valor numérico válido para o preço.
Preço: R$ 5.50
Quantidade em estoque: 20
Item cadastrado com sucesso!

1 - Cadastrar item
2 - Consultar itens
0 - Sair
Escolha: 1

Cadastrar novo item do cardápio
Nome do item: coca cola
Descrição: zero acucar
Preço: R$ 10
Quantidade em estoque: 50
Item cadastrado com sucesso!

1 - Cadastrar item
2 - Consultar itens
0 - Sair
Escolha: 2

--- Cardápio ---
Código: 1, Nome: Pão de forma, Descrição: integral, Preço: R$5.5, Estoque: 20
Código: 2, Nome: coca cola, Descrição: zero acucar, Preço: R$10.0, Estoque: 50
```

Figura 1. Cadastro e consulta de itens

```
1 - Criar pedido
2 - Ver pedidos pendentes
0 - Sair
Escolha: 1

Itens disponíveis:
Código: 1, Nome: Pão de forma, Preço: R$5.5, Estoque: 20
Código: 2, Nome: coca cola, Preço: R$10.0, Estoque: 50
Digite o código do item que deseja adicionar: 2
Item 'coca cola' adicionado ao pedido.
Deseja adicionar outro item? (s/n): n
Deseja aplicar cupom de desconto (%)? Caso não, pressione enter:

Pedido 2 criado com sucesso! Total: R$10.00
```

Figura 2. Criando pedido

```
1 - Criar pedido
2 - Ver pedidos pendentes
0 - Sair
Escolha: 2
Código: 1, Total: R$13.95, Status: AGUARDANDO APROVACAO
Código: 2, Total: R$10.00, Status: AGUARDANDO APROVACAO
```

Figura 3. Pedidos pendentes

```
1 - Processar pedidos pendentes
2 - Ver pedidos aceitos
0 - Sair
Escolha: 1

Pedido 1 - Total: R$13.95 - Status: AGUARDANDO APROVACAO
Aceitar ou Rejeitar o pedido? (a/r): r
Pedido 1 REJEITADO.

Pedido 2 - Total: R$10.00 - Status: AGUARDANDO APROVACAO
Aceitar ou Rejeitar o pedido? (a/r): a
Pedido 2 ACEITO.

1 - Processar pedidos pendentes
2 - Ver pedidos aceitos
0 - Sair
Escolha: 2
Código: 2, Total: R$10.00, Status: ACEITO
```

Figura 4. Processar, aceitar e recusar pedidos pendentes

4.3 Fluxograma do Funcionamento

O fluxo do programa pode ser representado conforme o diagrama abaixo:

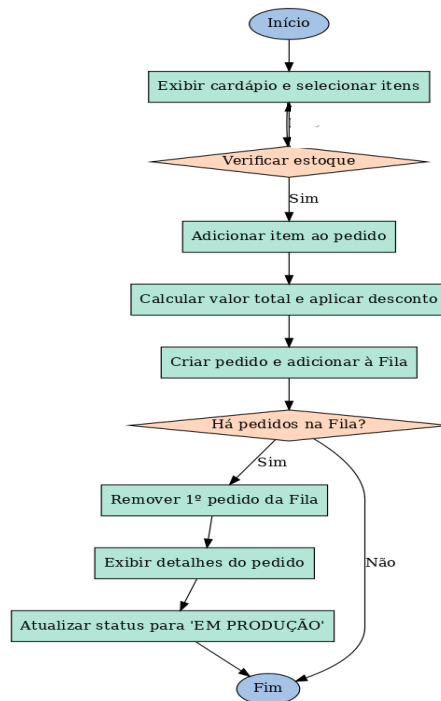


Figura 5. Fluxograma

5. Considerações finais

Esse projeto foi uma ótima oportunidade para colocar em prática o que pouco sabíamos e acabamos aprendendo a usar listas para organizar o cardápio e filas para gerenciar os pedidos. Mostrou na prática como essas estruturas ajudam a deixar o sistema mais organizado e funcional, garantindo que os pedidos sejam atendidos na ordem certa.

Trabalhar em Python, direto no console, facilitou bastante o foco no essencial: aplicar os conceitos sem depender de banco de dados ou de bibliotecas extras. Além disso, o código ficou de um jeito que, no futuro, dá para melhorar e até integrar com outras ferramentas, como um sistema de banco de dados ou até uma interface gráfica.

O maior desafio foi nos familiarizar com as ferramentas Git e GitHub. Tivemos que aprender a usar os comandos e organizar o repositório, já que não tínhamos esse conhecimento prévio. Mas foi um processo muito importante, que nos ajudou a desenvolver mais uma habilidade que é de extrema relevância para nossa formação.

No geral, esse trabalho ajudou a entender melhor como a escolha das estruturas de dados certas faz diferença na eficiência de um sistema. Mais do que reforçar a teoria, ele serviu para treinar lógica, programação e organização, habilidades que vão ser muito úteis na nossa formação em Sistemas de Informação.

6. Referencias

GODOY, Vinícius. **Estruturas de Dados**. IESDE BRASIL SA. 2020. Disponível em: https://books.google.com.br/books?hl=pt-BR&lr=&id=apVjEQAAQBAJ&oi=fnd&pg=PP2&dq=Introdu%C3%A7%C3%A3o+%C3%A0s+estruturas+de+dados+de+fila+e+suas+aplica%C3%A7%C3%B5es.&ots=-7cZERx-T9&sig=hrMzlOrhZzH6GYGz_X-5YRO-BAA&redir_esc=y#v=onepage&q=Introdu%C3%A7%C3%A3o%20%C3%A0s%20estruturas%20de%20dados%20de%20fila%20e%20suas%20aplica%C3%A7%C3%B5es.&f=false. Acesso em: 23 de set de 2025.

GUY, Uriel. **Apostila de Estruturas de Dados**. 2002. Disponível em: https://d1wqtxts1xzle7.cloudfront.net/56324257/apostila_estruturas_dados-libre.pdf?1523810725=&response-content-disposition=inline%3B+filename%3DApostila_de_Estruturas_de_Dados.pdf&Expires=1758858861&Signature=BDwq~Qg2K8D3hm8bKI5~yM7Kf3Hh8f4~LnyBJUw15ZY~m0QiLBj0OGNTcWNak9K4r1~wlicLoSISBu-8VTJp51l9LefXmks8MEvyQp5dJUy-z31k9YoQHxF5G6DRtLG4jLlpywowZHsuLFo uFdo01V~VZ1MGrmWUBaL~g1Mcj-hysGf-2KJFq3RtZX3Hm5FgVRPyKWakTZS9kU0JcEI8ljPKiT~S3hCRceCQC3hSRFEkEl48Hv78k4iqVCCWZbmltZ2aGsiR4tPQb1xf4fL9SomdMJVwzJKDHcD8jJC8P8HJWrvuinjtGwsILedoSx2vHSeEbRfXBRQcPiAmRew__&Key-Pair-Id=APKAJLOHF5GGSLRBV4ZA. Acesso em: 25 de set de 2025.

COSTA, Roselene Henrique Pereira et al. **Algoritmos e Estrutura de Dados**. 2023. Disponível em: <https://repositorio.pgsscogna.com.br/bitstream/123456789/52931/1/Algoritmos%20e%20Estrutura%20de%20Dados.pdf>. Acesso em: 25 de set de 2025.