

GCT053- Estruturas de Dados

Professor: Dr Johnatan Alves de Oliveira
Instituto de Ciências, Tecnologia e Inovação (ICTIN)
Universidade Federal de Lavras

2024-1

Vetor, Matriz, Lista ,Fila e Pilha

Pontos: 10
Preparação para Prova I

Data de entrega: 09/05/2024 23:59

Detalhes do Trabalho

1. O código-fonte deve ser desenvolvido em C;
2. A sua aplicação deve usar os conceitos estudados na disciplina;
3. Uso obrigatório das estruturas de dados estudadas: Vetor, Matriz, Lista, Fila e Pilha;
4. Todos os requisitos do trabalho devem ser cumpridos.
5. A entrega deverá acontecer na UFLA virtual. Todo o código-fonte deve ser enviado para ser avaliado. Não é permitido o uso de bibliotecas de terceiros;
6. Teste o seu código em outro computador ou site que execute código-fonte para garantir que não tenha algum erro de execução.
7. A entrega e condução do trabalho é individual, mas os alunos podem trocar informações entre si.

Contexto Geral

Você é um desenvolvedor de software encarregado de criar sistemas de gerenciamento para diferentes aplicações. Cada questão abaixo representa um desafio que deve ser resolvido com o uso adequado de estruturas de dados, mas cabe a você determinar qual estrutura é mais eficiente para cada situação.

Questão 01: Gerenciamento de Reservas de Hotel

No competitivo setor hoteleiro, onde a experiência do hóspede é primordial, a gestão eficiente das reservas é um fator chave para o sucesso. Um hotel com 100 quartos distribuídos por 10 andares enfrenta o desafio constante de maximizar a ocupação, enquanto oferece um serviço ágil e personalizado. Neste contexto, a capacidade de gerenciar reservas de forma eficaz é essencial para manter uma operação harmoniosa e rentável.

Com isso em mente, é essencial desenvolver um sistema de gerenciamento de reservas que seja robusto e intuitivo. O sistema deve ser capaz de registrar novas reservas, cancelar reservas existentes e verificar a disponibilidade dos quartos em tempo real. Além disso, é crucial que o sistema forneça uma visão geral instantânea da ocupação do hotel, permitindo que a equipe do hotel tome decisões informadas rapidamente. Implementar um sistema com essas capacidades não só melhora a eficiência operacional, mas também eleva a satisfação do cliente, garantindo que cada hóspede tenha uma experiência memorável e sem complicações.

Menu

Você deve utilizar o menu a seguir:

```
printf("\nMenu de Gerenciamento de Reservas de Hotel\n");
printf("1. Reservar um quarto\n");
printf("2. Cancelar uma reserva\n");
printf("3. Verificar disponibilidade de um quarto\n");
printf("4. Sair\n");
printf("Escolha uma opção: ");
```

Requisitos

Diante do que foi apresentado, o seu sistema deve cumprir os requisitos apresentados a seguir:

1. Registro de Reservas

Propósito: Permitir que a recepção e outros pontos de serviço ao cliente possam reservar quartos rapidamente, garantindo que os hóspedes recebam confirmações imediatas de suas acomodações.

Detalhes: Cada quarto deve poder ser reservado individualmente, indicando a data de check-in e check-out, e uma vez reservado, o quarto não deve estar disponível para outras reservas nas datas especificadas.

2. Cancelamento de Reservas

Propósito: Facilitar o cancelamento de reservas eficientemente para permitir a reutilização imediata do quarto, otimizando a taxa de ocupação do hotel.

Detalhes: O sistema deve permitir que as reservas sejam canceladas até um determinado período antes do check-in, liberando o quarto para ser reservado novamente sem atrasos.

3. Verificação de Disponibilidade

Propósito: Oferecer uma ferramenta rápida e fácil para verificar a disponibilidade de quartos, ajudando a equipe a responder às solicitações dos clientes em tempo real.

Detalhes: O sistema deve fornecer uma maneira clara de visualizar quais quartos estão disponíveis ou ocupados em qualquer data, através de uma interface gráfica ou relatórios detalhados.

4. Visão Geral da Ocupação

Propósito: Dar uma visão geral rápida da ocupação do hotel, permitindo à gerência do hotel e à equipe de planejamento ajustar estratégias de marketing e preços com base na ocupação atual e futura.

Detalhes: O sistema deve ser capaz de gerar visões gerais de ocupação que podem ser visualizadas em formatos diários, semanais ou mensais, ajudando na tomada de decisões estratégicas.

Exemplos de Entrada e Saída

1. Fazer uma Reserva

Escolha uma opção: 1

Data de entrada (DD/MM/AAAA): 15/06/2024

Data de saída (DD/MM/AAAA): 20/06/2024

Número de hóspedes: 2

Quartos disponíveis: 101, 102, 103...

Escolha um quarto: 101

Reserva confirmada para quarto 101 de 15/06/2024 até 20/06/2024 para 2 hóspedes.

Número da reserva:12345

2. Cancelar uma reserva

Escolha uma opção: 2

Informe o número da reserva: 12345

Reserva número 12345 cancelada com sucesso.

3. Verificar disponibilidade dos quartos

Escolha uma opção: 3

Data para verificar disponibilidade (DD/MM/AAAA): 25/12/2024

Quartos disponíveis em 25/12/2024: 105, 106, 107...

4. Sair

Escolha uma opção: 4

Obrigado por usar nosso sistema de gerenciamento de reservas!

Questão 02: Sistema de Controle de Estoque

Pequenos comércios, como lojas de varejo, são o coração do comércio local, oferecendo uma gama diversificada de produtos e contribuindo para a economia da comunidade. No entanto, esses estabelecimentos enfrentam desafios significativos na gestão de seus inventários. A capacidade de manter um equilíbrio entre estoque excessivo e insuficiente é crucial para evitar perdas financeiras e garantir a satisfação do cliente. Além disso, o rastreamento eficiente de vendas e reposições é fundamental para manter a operação fluindo sem interrupções.

Diante dessas necessidades, surge a importância de desenvolver um sistema de controle de estoque adaptado para atender às especificidades de um pequeno comércio. Este sistema deve ser capaz de lidar com a variedade de produtos e suas demandas únicas de reposição e rotatividade. Deve também fornecer ferramentas para monitorar as vendas e gerenciar reposições de forma ágil, permitindo que o comércio responda prontamente às flutuações do mercado. Implementar um sistema assim não só otimiza a gestão do inventário, mas também fortalece a capacidade do comércio de se adaptar às tendências de consumo, maximizando assim suas oportunidades de sucesso no mercado competitivo.

1. Adicionar Produtos ao Estoque

Propósito: Facilitar a inclusão de novos produtos no sistema, permitindo que o comércio acompanhe novos itens assim que eles chegam.

Detalhes: Cada novo produto será cadastrado com um código único, nome do produto, e a quantidade inicial em estoque. O sistema deve permitir a fácil adição de informações de produto e ajuste de inventário inicial.

2. Remover Produtos do Estoque

Propósito: Gerenciar a saída de produtos do estoque, seja por vendas ou por perda/expiração, assegurando que o inventário esteja sempre atualizado.

Detalhes: O sistema deve permitir a remoção de uma quantidade especificada de produtos do estoque, ajustando automaticamente o inventário e registrando a saída de itens.

3. Verificar a Quantidade Disponível de Cada Produto

Propósito: Oferecer à equipe de vendas informações em tempo real sobre a disponibilidade de produtos, ajudando a evitar vendas de itens esgotados e a gerenciar expectativas de clientes.

Detalhes: O sistema deve fornecer uma consulta rápida pelo código ou nome do produto para mostrar a quantidade disponível em estoque, facilitando a gestão de vendas e pedidos.

Menu

Você deve utilizar o menu a seguir:

```
printf("\nMenu de Gerenciamento de Estoque\n");
printf("1. Adicionar Produto\n");
printf("2. Remover Produto\n");
printf("3. Verificar Produto\n");
printf("4. Sair\n");
printf("Escolha uma opção: ");
```

Exemplos de Entrada e Saída

1. Adicionar Produto

```
Escolha uma opção: 1
Código do Produto: 001
Nome do Produto: Caneta Azul
Quantidade Inicial: 100
Produto adicionado com sucesso: Caneta Azul, Código 001, Quantidade: 100
```

2. Remover Produto

```
Escolha uma opção: 2
Código do Produto: 001
Quantidade a remover: 20
Produto removido com sucesso: Caneta Azul, Código 001, Quantidade removida: 20
```

3. Verificar Produto

```
Escolha uma opção: 3
Código do Produto: 001
Quantidade disponível: 80
```

4. Sair

Escolha uma opção: 4

Obrigado por usar nosso sistema de gerenciamento de estoque!

Questão 3: Histórico de Navegação

Na era digital atual, onde a internet se tornou uma extensão do espaço pessoal e profissional, a capacidade de rastrear e gerenciar o histórico de navegação web é mais importante do que nunca. Com a quantidade crescente de informações acessadas diariamente, os usuários precisam de ferramentas que não apenas armazenem seus dados de navegação, mas também ofereçam controle e flexibilidade sobre essas informações.

Neste contexto, surge a necessidade de desenvolver um sistema avançado de gerenciamento de histórico de navegação web. Este sistema deve ser projetado para registrar de forma precisa cada site visitado pelo usuário, seguindo uma ordem cronológica. Deve incluir uma funcionalidade de “desfazer”, permitindo aos usuários remover o último site visitado com um simples comando, e uma opção para visualizar o histórico de navegação de maneira reversa, do mais recente ao mais antigo. O objetivo é proporcionar uma experiência de navegação mais organizada e eficiente, permitindo aos usuários revisitar sites de interesse e gerenciar seu histórico digital com facilidade e precisão. A implementação desse sistema não só melhora a usabilidade e a experiência do usuário, mas também reforça a privacidade e a segurança dos dados pessoais na web.

1. Registrar Visitas a Websites **Propósito:** Capturar e armazenar cada site visitado pelo usuário em ordem cronológica, assegurando que todas as sessões de navegação sejam rastreadas adequadamente.

Detalhes: Cada entrada no histórico deve incluir o URL do site e o timestamp da visita. A adição de novos sites ao histórico deve ser eficiente, mesmo com um grande volume de dados.

2. Remover o Último Site Visitado

Propósito: Permitir aos usuários desfazer a última ação de navegação, simulando o botão “voltar” encontrado nos navegadores web.

Detalhes: Essa funcionalidade deve permitir a remoção do site mais recente do histórico, facilitando a correção de erros de navegação ou a revisão de informações.

3. Visualizar o Histórico de Navegação de Forma Reversa

Propósito: Proporcionar uma visão do histórico de navegação do mais recente ao mais antigo, auxiliando os usuários a relembrar e acessar sites que foram visitados anteriormente na mesma sessão.

Detalhes: O sistema deve conseguir apresentar todo o histórico em ordem inversa sem afetar o desempenho, mesmo que o histórico contenha um grande número de entradas.

Menu

Você deve utilizar o menu a seguir:

```
printf("\nMenu de Histórico de Navegação\n");  
printf("1. Visitar um novo site\n");  
printf("2. Voltar\n");  
printf("3. Exibir histórico de navegação\n");  
printf("4. Sair\n");  
printf("Escolha uma opção: ");
```

Exemplos de Entrada e Saída

Dica para pegar data e hora em C.[Clique aqui](#)

1. Visitar um novo site

Escolha uma opção: 1
URL do site: <https://www.example.com>
Site <https://www.example.com> visitado em 01/01/2024 às 12:00.

2. Voltar

Escolha uma opção: 2
Última visita removida: <https://www.example.com>

3. Exibir histórico de navegação

Escolha uma opção: 3
Histórico de navegação:
- <https://www.secondsite.com> visitado em 01/01/2024 às 11:50
- <https://www.othersite.com> visitado em 01/01/2024 às 11:30
- <https://www.startsite.com> visitado em 01/01/2024 às 11:00

4. Sair

Escolha uma opção: 4
Obrigado por usar nosso sistema de histórico de navegação!

Questão 4: Filas de Atendimento ao Cliente

Em um cenário de suporte técnico, onde a rapidez na resolução de problemas é essencial e as demandas dos clientes são diversas e muitas vezes críticas, a excelência no atendimento se torna um pilar central. A gestão eficiente da fila de atendimento emerge como um elemento vital para assegurar a contentamento dos clientes e a produtividade das operações. Um sistema meticulosamente elaborado para o gerenciamento da fila não apenas diminui significativamente o tempo de espera, mas também estrutura o processo de atendimento de maneira inteligente, garantindo que os recursos técnicos sejam utilizados da maneira mais eficaz possível. Isso não só melhora a experiência do cliente, mas também potencializa a utilização dos técnicos, reduzindo períodos ociosos e maximizando a resolução de problemas em tempo hábil. A implementação de tal sistema é, portanto, imperativa para sustentar altos padrões de serviço e fortalecer a fidelidade do cliente.

1. Adicionar Clientes à Fila

Propósito: Registrar cada novo pedido de suporte à medida que chega, garantindo que todos os clientes sejam atendidos de acordo com a ordem de chegada.

Detalhes: O sistema deve permitir que os atendentes insiram os detalhes do cliente (nome, número de contato, descrição breve do problema) e que esses detalhes sejam adicionados ao final da fila de atendimento.

2. Atender o Próximo Cliente

Propósito: Facilitar o atendimento sequencial, assegurando que o cliente que está há mais tempo esperando seja atendido primeiro.

Detalhes: O sistema deve permitir aos técnicos de suporte selecionar o próximo cliente a ser atendido, automaticamente removendo esse cliente da fila após o início do atendimento.

3. Exibir Todos os Clientes na Fila

Propósito: Oferecer aos gerentes e técnicos uma visão clara de todos os clientes atualmente esperando por atendimento, permitindo uma melhor alocação de recursos e estimativa de tempo de espera.

Detalhes: O sistema deve fornecer uma lista detalhada de todos os clientes na fila, mostrando suas informações e o tempo que já passaram na fila, ajudando na gestão e priorização de casos urgentes.

Menu

Você deve utilizar o menu a seguir:

```
printf("\nMenu de Gerenciamento de Fila de Atendimento ao Cliente\n");
```



```
printf("1. Adicionar Cliente à Fila\n");
printf("2. Atender o Próximo Cliente\n");
printf("3. Exibir Todos os Clientes na Fila\n");
printf("4. Sair\n");
printf("Escolha uma opção: ");
```

Exemplos de Entrada e Saída

1. Adicionar Clientes à Fila

Propósito: Registrar cada novo pedido de suporte assim que chega, garantindo que todos os clientes sejam atendidos em ordem de chegada.

Escolha uma opção: 1
Nome do Cliente: João Silva
Número de Contato: 123456789
Descrição do Problema: Problema com conexão de internet.
Cliente adicionado à fila com sucesso.

2. Atender o Próximo Cliente

Propósito: Assegurar que o cliente que está há mais tempo esperando seja atendido primeiro.

Escolha uma opção: 2
Atendendo o próximo cliente: João Silva
Problema: Problema com conexão de internet.

3. Exibir Todos os Clientes na Fila

Propósito: Oferecer uma visão clara de todos os clientes atualmente esperando por atendimento.

Escolha uma opção: 3
Clientes na fila:
1. Maria Oliveira - Espera: 15 minutos - Problema: Falha no software.
2. Carlos Andrade - Espera: 10 minutos - Problema: E-mail não funciona.

4. Sair

Escolha uma opção: 4
Obrigado por usar nosso sistema de gerenciamento de filas. Até mais!