

Estrutura de Dados - Lista 3 : Listas Encadeadas

Prof. Rodrigo de Sapienza Luna

Turma 1

11 de outubro de 2024

Questões e Exercícios sobre Lista Encadeada

Exercícios Básicos

1. Contar Elementos:

Implemente um método na classe `ListaEncadeada` que conte e retorne o número de elementos na lista.

2. Buscar Elemento:

- Crie um método que busque um elemento na lista e retorne `true` se ele existir ou `false` caso contrário.

3. Inverter a Lista:

- Escreva uma função que inverta a ordem dos elementos na lista encadeada.

4. Adicionar ao Fim:

- Modifique a classe para adicionar um novo método que insira um elemento no final da lista.

Exercícios Intermediários

1. Remover Duplicatas:

- Implemente um método que remova todos os elementos duplicados da lista.

2. Mesclar Duas Listas:

- Escreva uma função que recebe duas listas encadeadas e as mescle em uma terceira lista, mantendo a ordem.

3. Ordenar a Lista:

- Crie um método que ordene os elementos da lista encadeada usando o método de ordenação por seleção (selection sort) ou outro método de sua escolha.

Exercícios Avançados

1. Encontrar o N-ésimo Elemento do Final:

- Implemente um método que encontre o n-ésimo elemento a partir do final da lista (por exemplo, o 1º elemento a partir do final).

2. Remover N-ésimo Elemento:

- Escreva um método que remova o n-ésimo elemento da lista encadeada.

3. Duplicar a Lista:

- Crie uma função que duplique cada elemento da lista, inserindo um segundo nó com o mesmo valor após cada nó original.

4. Reverter a Lista Recursivamente:

- Implemente um método que inverta a lista encadeada usando recursão.

Exercícios Desafiadores

1. Palíndromo:

- Crie um método que verifique se os elementos da lista formam um palíndromo (ou seja, a lista é igual de frente para trás).

2. Remover Elementos em Posições Específicas:

- Implemente um método que remova elementos das posições especificadas (por exemplo, remova os elementos nas posições 1, 3 e 5).

3. Detectar Ciclos:

- Implemente um método que verifique se a lista contém um ciclo (ou seja, se um nó aponta de volta a um nó anterior).

4. Dividir a Lista em Duas Metades:

- Escreva uma função que divida a lista encadeada em duas metades e retorne ambas as metades como duas novas listas.