



UNIVERSIDADE FEDERAL DO CEARÁ
CAMPUS CRATEÚS
SISTEMAS DE INFORMAÇÃO
DOCENTE: RAFAEL MARTINS BARROS
DISCIPLINA: ESTRUTURA DE DADOS

DISCENTES: ANA JAZA TAVARES SOARES - 558540, ANA LETICIA DE OLIVEIRA
SOARES - 555346.

RELATÓRIO: LISTA LINEAR SEQUENCIAL

SUMÁRIO

1. Objetivo.....	2
2. Principais funções implementadas e notação Big-O.....	2
2.1. Criar Lista.....	2
2.2. Inserir Elemento no Final.....	2
2.3. Inserir Elemento em uma Posição.....	3
2.4. Remover Elemento por Posição.....	3
2.5. Remover Elemento por Valor.....	3
2.6. Obter Elemento por Posição.....	3
2.7. Procurar Elemento.....	3
2.8. Tamanho da Lista.....	3
2.9. Imprimir Lista.....	3
3. Como o trabalho foi organizado.....	3
4. Principais dificuldades.....	4
5. Conclusão.....	4

1. Objetivo

O objetivo deste trabalho foi implementar uma lista linear sequencial em linguagem C, aplicando os conceitos de estrutura de dados. O projeto foi desenvolvido em dupla pela Jaza e pela Letícia, seguindo uma abordagem incremental, realizando commits no GitHub.

A lista linear sequencial foi implementada utilizando um vetor estático, com operações como inserção, remoção, busca e exibição de elementos. A escolha por um vetor estático se deu pela simplicidade na gestão de memória, uma vez que o tamanho da lista foi pré-definido.

2. Principais funções implementadas e notação Big-O

2.1. Criar Lista

Inicializa uma nova lista vazia.

$O(1)$, só atribui zero ao valor do tamanho pois a lista tem um tamanho fixo correspondente a constante $MAX(100)$.

2.2. Inserir Elemento no Final

Adiciona um elemento inteiro ao final da lista.

$O(n)$ no pior caso tem que percorrer os n elementos da lista para poder fazer a inserção.

2.3. Inserir Elemento em uma Posição

Adiciona um elemento inteiro em uma posição específica da lista.

$O(1)$ pois como já se sabe a posição basta atribuir se o espaço estiver disponível.

2.4. Remover Elemento por Posição

Remove o elemento na posição especificada.

$O(1)$ pois como já se sabe a posição basta zerar o elemento especificado.

2.5. Remover Elemento por Valor

Remove a primeira ocorrência de um elemento com um valor específico.

$O(n)$ pois no pior caso o elemento está na posição n .

2.6. Obter Elemento por Posição

Retorna o elemento na posição especificada.

$O(1)$, basta retornar o elemento do índice diretamente

2.7. Procurar Elemento

Retorna a posição da primeira ocorrência de um valor especificado. Se não encontrado, retorna uma indicação de não encontrado.

$O(n)$, no pior caso ele percorre a lista toda.

2.8. Tamanho da Lista

Retorna o número de elementos atualmente na lista.

$O(1)$, basta retornar o valor da variável tamanho

2.9. Imprimir Lista

Exibe todos os elementos da lista em ordem.

$O(n)$, percorre os n elementos da lista mostrando os elementos.

3. Como o trabalho foi organizado

Para a realização deste trabalho, foi formado um grupo na plataforma de mensagens Whatsapp - focado em discutir e buscar soluções para cada tarefa solicitada. Com o objetivo de manter uma boa organização e divisão de responsabilidades, foram divididas as implementações pela metade. A Ana Letícia ficou responsável pelas cinco primeiras funções, enquanto a Jaza assumiu as funções restantes, como também, a criação de um menu que permite ao usuário uma interação eficiente com as funcionalidades.

4. Principais dificuldades

A principal dificuldade foi imaginar os comportamentos da estrutura de dados para que se fizesse a implementação. Diante disso, foi feito o estudo do slide e foram feitos alguns desenhos para facilitar a elucidação dos comportamentos das funcionalidades da estrutura de dados. Seguir esses passos foi de extrema importância no processo de aprendizagem e fixação dos conteúdos apresentados em sala de aula.

5. Conclusão

Durante o desenvolvimento, foram realizados diversos testes para verificar se as operações da lista funcionam corretamente. Casos de teste incluíram inserções e remoções em diferentes posições, busca de elementos existentes e inexistentes, além de testes para garantir o comportamento adequado da lista quando cheia.

O projeto foi concluído com sucesso, atendendo aos requisitos estabelecidos. O uso do GitHub foi essencial para o controle das versões e o registro do progresso do desenvolvimento.