

Atividade Prática 02

“Listas Duplamente encadeadas”

Universidade Tecnológica Federal do Paraná (UTFPR), campus Apucarana
Curso de Engenharia de Computação
Disciplina de Estrutura de Dados - ED62A - 1º Semestre 2019
Prof. Dr. Rafael Gomes Mantovani

1 Descrição

Elabore um programa em C que implemente um tipo abstrato de dados para lista duplamente encadeada, e suas operações de manipulação. Uma lista duplamente encadeada é um arranjo de dados onde cada elemento é também um tipo abstrato de nó de lista (**NoLista**) que guarda dois ponteiros, como mostrado na Figura 1:

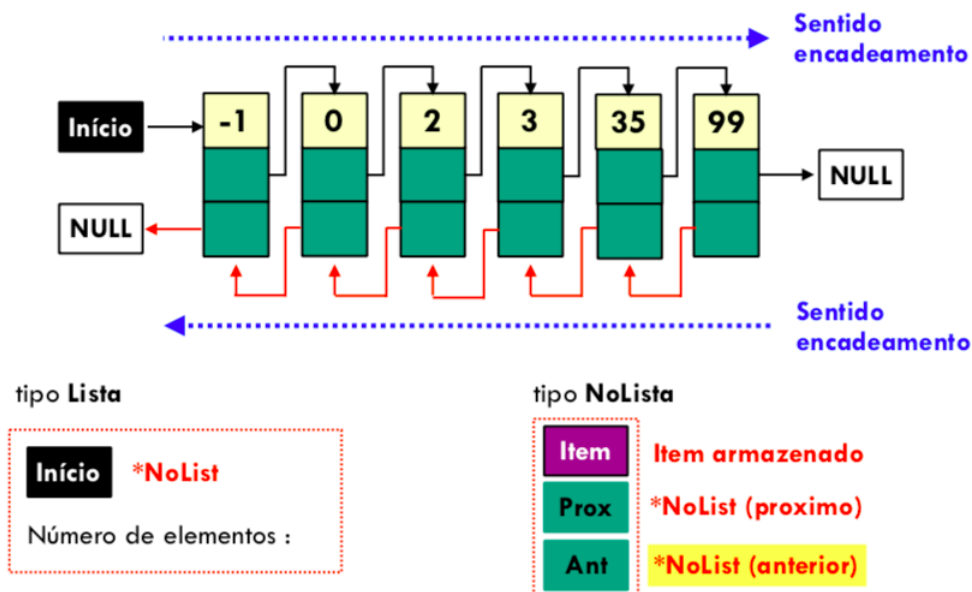


Figura 1: Diagrama representativo de uma lista duplamente encadeada com 6 elementos. No diagrama é possível também ver uma representação gráfica dos tipos abstratos de dados envolvidos em sua codificação.

- **anterior**: ponteiro que aponta para o elemento anterior na lista ordenada;
- **próximo**: ponteiro que aponta para o elemento posterior na lista ordenada.

Use as implementações das estruturas já desenvolvidas em sala para iniciar sua implementação. Além da definição e codificação dos tipos de dados, a estrutura implementada deve ser capaz de:

1. ser iniciada;
2. ter seu tamanho retornado;
3. verificar se está vazia (ou não);
4. inserir um elemento de acordo com um valor passado por parâmetro (chave);
5. pesquisar se um elemento pertence ou não a estrutura;
6. remover um elemento dado um valor de consulta (chave);
7. remover o último elemento da lista ordenada;
8. remover o primeiro elemento da lista ordenada;
9. imprimir os valores armazenados na lista;
10. retornar o valor do primeiro elemento;
11. retornar o valor do último elemento;
12. destruir a lista, desalocando a memória de cada nó existente;

Obs: no programa principal (`main.c`) inserir comandos que testem todas as operações da estrutura.

2 Orientações gerais

- Implementar também o controle de erros, para lidar com exceções que possam ocorrer;
- Para acompanhamento do desenvolvimento, criar um repositório individual com o código desenvolvido no `github Classroom`, por meio do link: <https://classroom.github.com/a/rYDj3fVb>. Os repositórios serão privados, com acesso apenas do professor e do aluno.
- Entrega do programa final: via Moodle. O aluno deve submeter o fonte no link da atividade disponibilizado na página da disciplina no Moodle.
- **Data de entrega:** 02/05/2019.
- Os códigos desenvolvidos por cada aluno serão também verificados por ferramentas de plágio. Códigos iguais/similares terão nota zero.

Referências

- [1] Thomas H. Cormen,; Ronald Rivest; Charles E. Leiserson; Clifford Stein. Algoritmos - Teoria e Prática - 3ª Ed. Elsevier - Campus, 2012.
- [2] Nivio Ziviani. Projeto de algoritmos com implementações: em Pascal e C. Pioneira, 1999.
- [3] Adam Drozdek. Estrutura De Dados E Algoritmos Em C++. Cengage, 2010.