

UENF

Universidade Estadual do Norte Fluminense Darcy Ribeiro

Curso: Ciência de Computação

Data: 16/05/2022

Atividade: LE2

Período: 3º

Disciplina: Estrutura de Dados I

Professor: Fermín Alfredo Tang

Turno: Diurno

Nome do aluno: Mariana Cossetti Dalfior e Enzo Picanço Alberoni

Matrícula: 20211100064, 20211100053

Questões da LE2

1.- **(1,5 Pontos)** Implementar uma **pilha** como TAD considerando as seguintes variantes:

- i) Usando estrutura estática **específica** (usando inteiros) que contenha um vetor simples e um contador; Referência: Prog.3.2 Laureano ou Seção 8.4 Backes.
- ii) Usando estrutura dinâmica **específica** (usando inteiros) que use uma estrutura cabeçalho. Referência: Seção 8.5 ou 8.6 Backes.
- iii) Usando uma estrutura dinâmica **genérica** (usando qualquer dado). Conforme abordada em aula. Referência: Seção 3.4 Gilbert e Forouzan usado em aulas.

Teste todas as implementações na inversão de uma sequência numérica.

2.- **(2,0 Ponto)** Modifique o programa de casamento de parênteses para considerar o casamento de { }, [] e (). O programa deverá verificar a correspondência entre o símbolo de abertura e seu correspondente de fechamento. Identifique e reporte os tipos de erro possíveis caso existam, indicando a linha, ou uma mensagem indicando que o casamento é perfeito. Teste o seu programa com arquivos de texto, com erros e sem erros. Referência: Programa 3.17 Gilbert e Forouzan usado em aulas.

3.- **(1,5 Pontos)** Implementar uma **fila** como TAD considerando as seguintes variantes:

- i) Usando uma estrutura estática **específica** (usando inteiros) que contenha um vetor circular e um contador. Referência: Prog.4.4 Laureano
- ii) Usando uma estrutura dinâmica **específica** (usando inteiros) que use uma estrutura cabeçalho. Referência: Seção 6.5 ou 6.6 Backes.
- iii) Usando uma estrutura dinâmica **genérica** (usando qualquer dado). Conforme abordada em aula. Referência: Seção 4.3 Gilbert e Forouzan usado em aulas.

Teste todas as implementações com uma sequência numérica.

4.- **(1,0 Ponto)** Escreva um programa em C que classifique as palavras em um texto, disponível na forma de arquivo, em categorias de acordo com a letra inicial de cada palavra. As palavras iniciadas com a mesma letra deverão ser colocadas em uma mesma fila. Após a classificação, exiba o número de palavras existentes em cada categoria. Referência: Prog. 4.11 Gilbert e Forouzan usado em aulas.

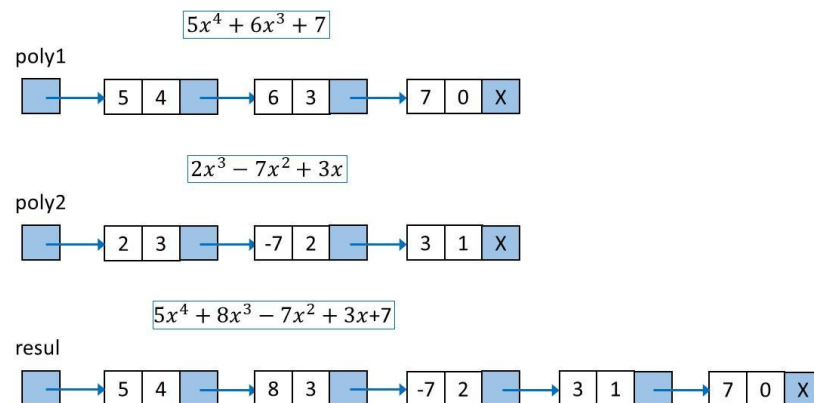
5.- **(1,0 Pontos)** Implementar uma lista ordenada como TAD considerando as variantes:

- i) Usando uma estrutura dinâmica **específica** (definida para inteiros) que use uma estrutura cabeçalho. Referência: Seção 5.5.2 Backes ou Programa 6.3 Laureano.

ii) Usando uma estrutura dinâmica **genérica** (que pode trabalhar com qualquer dado). Conforme abordada em aula. Referência: Seção 5.3 Gilbert e Forouzan usado em aulas.

X 6.- **(2,0 Pontos)** Escreva um programa que **some** polinômios. Cada polinômio deve ser representado como uma lista encadeada. O primeiro nó na lista representa o termo de maior grau, enquanto que o último representa o termo de menor grau. Cada nó contém três campos: i) coeficiente do termo; ii) expoente do termo; iii) o ponteiro ao seguinte termo.

Os polinômios podem ser adicionados utilizando um processo de intercalação das listas além da soma dos elementos com expoente igual. Observe o exemplo da Figura.



Regras para a adição:

- Se os expoentes são iguais, os coeficientes são somados e o resultado inserido;
- Se os expoentes são diferentes, o termo com maior expoente deverá ser inserido, enquanto o menor aguarda para ser comparado com outro termo.
- Se o resultado da adição é 0, o termo deve ser eliminado do polinômio soma.
- Se um polinômio termina antes do outro, os termos restantes do polinômio maior serão adicionados no polinômio soma.

Os dados do polinômio (coeficientes e potências) podem ser lidos de um arquivo ou ingressados do teclado. Imprima os dois polinômios de entrada e o polinômio soma.

X 7.- **(1,0 Ponto)** Modifique a TAD Lista de maneira a implementar uma **lista duplamente encadeada**. Inclua uma função para percorrer a lista no sentido inverso (backward). Teste a TAD duplamente encadeada usando os dados de censo populacional das cidades dos EUA. Crie uma lista duplamente encadeada com esses dados, ordenada com base no campo população. Imprima os dados da lista em ordem crescente (forward) e decrescente (backward). Referência: Seção 5.7 Backes ou adaptação de Seção 5.3 Gilbert e Forouzan

Metropolitan area	2000
New York–No. NJ	21,199,865
Los Angeles area	16,373,645
Chicago area	9,157,540
Washington–Baltimore	7,608,070
San Francisco area	7,039,362
Philadelphia–Atlantic City area	6,188,463
Boston area	5,819,100
Detroit area	5,456,428
Dallas–Fort Worth	5,221,801
Houston–Galveston area	4,669,571

Atlanta area	4,112,198
Miami–Fort Lauderdale	3,876,380
Seattle area	3,554,760
Phoenix area	3,251,876
Minneapolis–St. Paul	2,968,806
Cleveland area	2,945,831
San Diego area	2,813,833
St. Louis area	2,603,607
Denver area	2,581,506
San Juan, PR, area	2,450,292