

Lista 04

Listas: encadeamento simples e alocação dinâmica

- **Atenção:**

1. **Identificadores de variáveis:** escolha nomes apropriados;
2. **Documentação:** inclua cabeçalho, comentários e indentação no programa.
3. **Arquivo-base:** você deve usar o arquivo-fonte incompleto fornecido junto com a lista. É necessário completar as operações nos lugares indicados e você não deve realizar nenhuma alteração nas partes fornecidas. Inclusive, se houverem comandos de entrada (`scanf`) e saída (`printf`) definidos, estes não poderão ser alterados.

- **Exercícios:**

1. Faça um programa que receba os termos de dois polinômios (coeficiente e expoente) e que permita obter um terceiro polinômio (resultado) baseado em adição, subtração e multiplicação dos dois polinômios inicialmente recebidos, conforme as opções previamente configuradas no código-base deste exercício. O programa também deve ser capaz de realizar o cálculo de qualquer um dos polinômios do programa baseado no valor x também recebido como entrada. Você não pode alterar o procedimento principal, insira os códigos apenas nos lugares marcados. Opções do menu:
 0. **Sair:** libera a memória utilizada;
 1. **Adicionar(p, c, e):** adiciona ao polinômio p o termo $c.x^e$;
 2. **Remover(p, e):** remove do polinômio p o termo com expoente e ;
 3. **Somar(p_1, p_2):** retorna o polinômio $p_3 = p_1 + p_2$;
 4. **Subtrair(p_1, p_2):** retorna o polinômio $p_3 = p_1 - p_2$;
 5. **Multiplicar(p_1, p_2):** retorna o polinômio $p_3 = p_1 * p_2$;
 6. **Calcular polinômio(p, x):** retorna o valor polinômio p , usando o valor x .

Complete o arquivo L04EX01.c

Exemplos de E/S (os comentários entre parênteses não deverão ser exibidos):

Entrada	Saída
2	(Valor de $x = 2$)
1	
1 1 1	(Primeiro Polinômio: x)
1	
1 1 2	(Primeiro Polinômio: $x + x^2$)
1	
2 2 3	(Segundo Polinômio: $2x^3$)
2	
1 2	(Primeiro Polinômio: x)
3	(Resultado: $2x^3 + x$)
6	
1	2
4	
1 2	(Resultado: $x - 2x^3$)
6	
3	-14
5	(Resultado: $2x^4$)
6	
3	32
0	

2. Implemente uma lista de números inteiros com ordenação crescente (ou seja, a lista sempre armazena os elementos em ordem crescente) e sem repetição (ou seja, se tentarmos inserir elementos repetidos, estes não são realmente inseridos) com encadeamento simples e alocação dinâmica. Utilize estes procedimentos em um programa que receberá como entrada comandos de um menu de opções, conforme tabela a seguir. Este exercício não possui código inicial. Faça a sua documentação, incluindo o cabeçalho. Cada comando está associado a uma ação na lista e saída em formato específico.

0. **Sair**: libera a memória utilizada;
1. **Inserir(valor)**: insere o valor na posição correta da lista (mantendo a ordenação crescente);
2. **Procurar(valor)**: imprime na tela a posição do valor na lista (começando do zero), ou -1 caso não encontre;
3. **Remover(valor)**: remove da lista o valor;
4. **Imprimir()**: exibe uma linha na tela com todos os elementos da lista, seguidos por um espaço, na ordem crescente dos elementos. Se a lista estiver vazia, uma linha vazia é exibida.

Exemplos de E/S:

Entrada	Saída
1	
10	
1	
15	
1	
12	
2	
12	1
4	10 12 15
3	
13	
2	
15	2
1	
13	
4	10 12 13 15
3	
10	
4	12 13 15
1	
15	
4	12 13 15
2	
10	-1
0	