

Estrutura de Dados

Prof. Frederico S. Oliveira - Prazo final 04/02/2016

Segundo Semestre de 2016

Primeiro Trabalho Prático

1 Introdução

Faça um programa que realize a manipulação de dois polinômios com coeficientes inteiros. Um polinômio é constituído por um conjunto de termos. Cada um destes termos é formado por um número inteiro positivo n , incluindo o zero, que representa o grau e um número inteiro k que representa o coeficiente. Dessa forma, um polinômio é representado da seguinte forma:

$$k_n x^n + k_{n-1} x^{n-1} + k_{n-2} x^{n-2} + \dots + k_1 x + k_0$$

2 Desenvolvimento

Cada polinômio deve obrigatoriamente ser armazenado em uma LISTA formada por nós com alocação dinâmica. Cada nó da lista armazena os dados de um dos termos do polinômio: o coeficiente e o expoente, além do apontador para o próximo termo. Apenas termos com coeficiente não-nulo devem estar na lista. Exemplo: O polinômio $P(x) = -x^5 + 2x^3 + 5x^2 - 7$ poderia ser armazenado em uma lista L de acordo com o Tipo Abstrato de Dado (TAD) apresentado na figura (1).

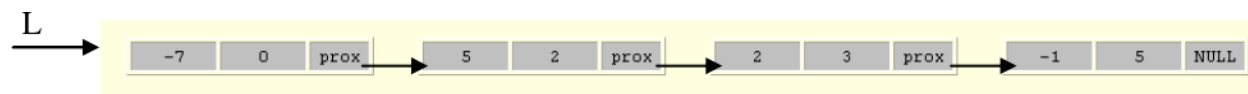


Figura 1: Exemplo de TAD para representar um polinômio.

A representação do polinômio não está totalmente restrita a esse TAD, sendo possível incrementá-la, conforme as necessidades do programa, desde que mantenha sua ideia original. Considerando a existência do TAD “poli” para representar polinômios, o seu programa deve disponibilizar as operações listadas na tabela (2). Nessa tabela, são apresentadas sugestões de protótipos das funções, que podem ser alteradas conforme a necessidade. As operações devem ser realizadas utilizando o TAD proposto.

Ao realizar a operação de impressão, os polinômios não podem ter mais do que um termo do mesmo grau, porque eles podem ser reduzidos a um só termo. Por exemplo, se os dois termos: $k_1 x^n$ e $k_2 x^n$ pertencem a um mesmo polinômio, então podem ser representados pelo termo: $(k_1 + k_2)x^n$

Para cálculo da derivada, observe que a derivada de um polinômio corresponde à soma das derivadas dos seus termos. Portanto, considere

$$P(x) = f_m(x) + f_{m-1} + \dots + f_1(x) + f_0(x),$$

Protótipo Função	Descrição
poli *criaPolinomio(int[])	Cria um novo polinômio. Recebe uma lista de inteiros contendo o grau e coeficiente de cada termo.
void apagaPolinomio(poli)	Apaga um polinômio, liberando toda a sua memória alocada.
void adicionaTermo(poli,int,int)	Adiciona um termo a um polinômio. Recebe dois inteiros, respectivamente, o coeficiente do termo e o seu grau.
void apagaTermo(poli,int)	Apaga um termo de um polinômio. Recebe um inteiro que indica o grau do termo a ser apagado.
int grauPolinomio(poli)	Retorna o grau do polinômio. O grau de um polinômio é o grau do termo de maior grau.
int coeficiente(poli,int)	Retorna o coeficiente de um termo de um polinômio. O inteiro representa o grau do termo pretendido.
poli *somaPolinomios(poli,poli)	Soma dois polinômios corresponde a somar os termos de igual grau de cada um dos polinômios: $k_1x + k_2x = (k_1 + k_2)x$. Retorna um novo polinômio.
poli *subPolinomios(poli,poli)	Subtrai dois polinômios corresponde a subtrair os termos de igual grau de cada um dos polinômios: $k_1x - k_2x = (k_1 - k_2)x$. Retorna um novo polinômio.
poli *multPolinomios(poli,poli)	Multiplica dois polinômios. Para isso, usa-se a propriedade distributiva: multiplica-se um polinômio pelos termos do outro. Retorna um novo polinômio.
poli *derivaPolinomio(poli)	Deriva um polinômio. Esta operação corresponde à derivação dos seus termos, melhor explicada abaixo. Retorna um novo polinômio.
float valorPolinomio(poli,float)	Calcula o valor do polinômio quando a variável x assume um dado valor, que é passado para a função como real.
void imprimePolinomio(poli)	Escreve o polinômio na tela, com seus termos ordenados em função de seu grau.

Tabela 1: Operações disponíveis para manipulação de polinômios.

então a derivada $P'(x)$ é definida da seguinte forma

$$P'(x) = f'_m(x) + f'_{m-1} + \dots + f'_1(x) + f'_0(x).$$

Observe que $P'(x)$ também é um polinômio. Ainda, para calcular a derivada $f'(x)$ do termo $f(x) = bx^m$ considere a seguinte estratégia:

- (i) Se $m \neq 0$ então $f'(x) = (m \times b)x^{m-1}$.
- (ii) Se $m = 0$ então $f'(x) = 0$.

3 Interface

A interface com o usuário deve ser em modo-texto. Inicialmente seu programa deve exibir informações básicas, conforme listado no código abaixo.

```
#####
# Trabalho Manipulacao de Polinomios
# Disciplina Estrutura de Dados
# Prof. Frederico S. Oliveira
# Autor: Fulano de Tal  RGA: 20159999999
#####
```

Em seguida, deve ser impresso um menu com informações para realizar as seguintes operações:

- a) Entrar com os dados dos dois polinômios, $P_1(x)$ e $P_2(x)$, a serem manipulados.
- b) Adicionar termo a um polinômio, solicitando o grau n e o coeficiente k desse termo.
- c) Apagar termo de um polinômio, solicitando o grau n do termo a ser removido.
- d) Verificar grau de um polinômio.
- e) Verificar coeficiente de um termo de um polinômio, solicitando o grau n do termo.
- f) Somar os polinômios $P_1(x)$ e $P_2(x)$.
- g) Multiplicar os polinômios $P_1(x)$ e $P_2(x)$.
- h) Derivar um polinômio.
- i) Calcular valor do polinômio com um valor real x .
- j) Sair do programa.

Quando necessário, seu programa deve solicitar mais informações do usuário, como por exemplo, a escolha de qual polinômio ($P_1(x)$ ou $P_2(x)$) manipular. O programa deve ser finalizado apenas quando o usuário informar a opção para sair.

4 Entrada

Ao informar os dados de um polinômio, o usuário digitará sequencialmente seus coeficientes e graus, ordenados em função do grau de cada termo. Por exemplo, o polinômio $P(x) = -x^5 + 2x^3 + 5x^2 - 7$ será digitado, em uma única linha, da seguinte forma:

-1 5 2 3 5 2 7 0

O polinômio $P(x) = x^{10} + 5x^5 + x^3 - 3x^2$ será digitado da seguinte forma:

1 10 5 5 1 3 -3 2

Nos testes serão fornecidos no máximo vinte termos.

5 Saída

Após executar cada uma das operações listadas na seção (3), com exceção da opção para sair, o programa deve imprimir o resultado. Por exemplo, ao receber do usuário o polinômio $P_1(x) = -x^5 + 2x^3 + 5x^2 - 7$, o mesmo deve ser impresso da seguinte forma:

P1(x)=-x^5+2x^3+5x^2-7

6 Documentação

Obrigatoriamente a documentação deve conter:

1. Introdução: descrição do problema a ser resolvido e visão geral sobre o funcionamento do programa.
2. Desenvolvimento: descrição sobre a implementação do programa. Deve ser detalhada a estrutura de dados utilizada (preferencialmente com diagramas ilustrativos), o funcionamento das principais funções e procedimentos utilizados, o formato de entrada e saída de dados, bem como decisões tomadas relativas aos casos e detalhes de especificação que por ventura estejam omissos no enunciado.
3. Compilação e Execução: descrição de como como compilar o código-fonte, bem como a sua execução.
4. Testes: descrição de quais testes foram executados e as saídas obtidas.
5. Conclusão: comentários gerais sobre o trabalho e as principais dificuldades encontradas em sua implementação.
6. Bibliografia: bibliografia utilizada para o desenvolvimento do trabalho, incluindo livros e/ou sites da internet se for o caso, seguindo padrão ABNT.

7 Material a ser Entregue

Os seguintes itens devem ser enviados via Ambiente Virtual de Aprendizagem (AVA), em um único arquivo compactado, com extensão “.tar.gz”:

- Código fonte do programa em C, bem indentado e comentado. O código-fonte, bem como os testes de sua execução, valem 60% da nota.
- Documentação do trabalho, que deve ser entregue mandatoriamente no formato PDF. A documentação vale 40% da nota.
- Caso também seja entregue a documentação em LaTeX, ganha-se 1 ponto, em um total de 10 pontos, não acumulativo.

8 Conclusão

Para cálculo da nota, serão realizados testes, e a nota será calculada com base nestes testes. Será atribuído zero aos seguintes casos:

- Código-fonte não compilar no Linux.
- Nenhum teste for executado com sucesso.
- Não apresentar documentação.
- Plágio.

Este trabalho deve ser desenvolvido de forma individual. Não serão realizadas exceções. Bom trabalho.