Trabalho 1
Cálculo Numérico
Programação de Computadores 1

Dhiego Loiola de Araújo   Daniel Saad Nogueira Nunes

Informações Preliminares

- Data limite de entrega: 20/09/2019.
- O trabalho deve ser feito **individualmente**.
- O aluno deverá inserir todos os recursos necessários para compreender e reproduzir os resultados em uma pasta zipada nomeada com o seguinte formato: `nome_sobrenome.zip`, em que `nome` é o nome do aluno e `sobrenome` o sobrenome.
- O arquivo zipado deverá ser entregue no seguinte formulário: `https://docs.google.com/forms/d/e/1FAIpQLSduBfCKJNT8iYveC9ZwUOOR4jIa-hzUb9VTuYTpgoswGh7k_Q/viewform?usp=sf_link`. **Observação**: é necessário ter um e-mail *Google* para conseguir fazer o *upload* do arquivo.
- Todas as atividades de programação devem ser realizadas com a linguagem *C*.
- Na ocorrência de **plágio**, todos os trabalhos envolvidos serão avaliados com nota **zero**.

1 Épsilon da Máquina

A representação de números reais é realizada através de uma aproximação pois a quantidade de bits utilizada na representação IEEE 754 é finita. Isto implica que não é possível representar todos os números reais em um sistema computacional com precisão infinita.

1.1 Objetivo

Utilizando alguma estrutura de repetição, crie um algoritmo que imprima na tela o menor número positivo em **ponto flutuante** e em **ponto flutuante precisão dupla** que a máquina considera diferente de zero.

Existem vários algoritmos para determinar o Épsilon da máquina, fica ao critério do estudante a pesquisa e implementação do método.

1.2 Entrada

O programa não possui entrada.

1.3 Saída

O programa deverá imprimir na primeira e segunda linhas, respectivamente, o menor número real positivo em ponto flutuante de precisão simples e o menor número real positivo em ponto flutuante de precisão dupla.

1.4 Instruções de Entrega

Um arquivo chamado `epsilon.c` deverá ser criado contendo a implementação acima e deve estar anexo a pasta a ser enviada via o formulário de entrega.

2 Polinômio de Taylor

Um polinômio de Taylor pode ser descrito através da definição abaixo.

Definição 2.1 (Polinômio de Taylor). *Seja f uma função n vezes diferenciável em um intervalo contendo a como ponto interior. O polinômio de Taylor P_n gerado por f é dado por:*

$$P_n(x) = f(a) + f'(a)(x-a) + \frac{f''(a)}{2!}(x-a)^2 + \cdots + \frac{f^{(n)}(a)}{n!}(x-a)^n \tag{1}$$

O ponto a é considerado o centro do polinômio. Este procedimento gera uma aproximação de funções através de polinômios e é utilizada com bastante frequência no dia a dia para obter valores numéricos de funções complexas através de operações básicas da aritmética, como soma e multiplicação.

Por exemplo, se $f(x) = 5x^2 - 2x^4 + 1.5x^3 - 10x^2 + 2x + 1$, temos que, quando $a = 0$, $P_2(x) = f(0) + f'(0) \cdot x + \frac{f''(0)}{2!} \cdot x^2$.

2.1 Objetivos

O objetivo desta tarefa de implementação é, dado um valor de x, computar os polinômios $P_i(x)$ com centro $a = 0$ e comparar com os valores de $f(x)$ até que $P_i(x)$ esteja suficientemente próximo de $f(x)$.

2.2 Entrada

A primeira linha da entrada possui um inteiro n ($0 \leq n \leq 6$) indicando o grau do polinômio.

A segunda linha da entrada possui $n + 1$ números reais $a_0, \ldots, a_n$ ($-10 \leq a_i \leq 10$), separados por espaço, que descrevem os coeficientes do polinômio

$$f(x) = a_0 + a_1x + a_2x^2 + \ldots + a_nx^n$$

A terceira linha da entrada, contém um inteiro q ($0 \leq q \leq 100$) indicando a quantidade de consultas.

As próximas q linhas cada, possuem um número real x, indicando que o polinômio de taylor deve ser avaliado sobre x.

2.3 Saída

Para cada um dos valores de x lidos, o seu programa deverá imprimir linhas no formato <i> <P_i(x)> <f(x)>, em que <i> é o índice do polinômio P_i, <P_i(x)> é o valor de $P_i(x)$ e f(x) o valor de $f(x)$. O seu programa deverá parar de imprimir assim que $|P_i(x) - f(x)| \leq 10^{-3}$. Após cada uma das impressões, seu programa deverá imprimir uma linha em branco.

2.4 Exemplo

Entrada

```
5
1 2 3 4 5 6
2
2.5
-2
```

Saída

```
1 1.000000 868.500000
2 6.000000 868.500000
3 24.750000 868.500000
4 87.250000 868.500000
5 282.562500 868.500000
6 868.500000 868.500000


1 1.000000 -135.000000
2 -3.000000 -135.000000
3 9.000000 -135.000000
4 -23.000000 -135.000000
5 57.000000 -135.000000
6 -135.000000 -135.000000
```

Entrada

```
2
-1 -2 1
2
0
-1
```

Saída

```
1 -1.000000 -1.000000

1 -1.000000 2.000000
2 1.000000 2.000000
```

```
3 2.000000 2.000000
```

2.5 Instruções de Entrega

Um arquivo chamado `taylor.c` deverá ser criado contendo a implementação acima e deve estar anexo a pasta a ser enviada via o formulário de entrega.

3 Método da Bisseção

Seja $f : [l, r] \to \mathbb{R}$ uma função contínua sobre o intervalo real $[l, r]$. Se $f(l)$ e $f(r)$ possuem sinais opostos, é possível concluir que existe ao menos uma raiz no intervalo $[l, r]$.

Partindo desta premissa, é possível aplicar um método de *divisão e conquista* denominado busca binária. Ele consiste no seguinte.

1. Seja l e r o intervalo que contém uma raiz de $f(x)$.
2. Enquanto a raiz não tiver sido encontrada ou o número máximo de iterações não tiver sido atingido:
   (a) Calcule m como sendo o ponto intermediário entre l e r, isto é, $m := l+(r-l)/2$.
   (b) Avalie $f(m)$, caso $|f(m)| < \epsilon$, termine o procedimento e dê como resposta a raiz m.
   (c) Caso $f(m)$ e $f(l)$ possuam o mesmo sinal, faça $l := m$.
   (d) Caso contrário, atualize $r := m$.
3. Se o número máximo de iterações foi atingido, reporte que a raiz não foi encontrada.

Quando aplicado em uma função sobre os reais, a busca binária também é conhecida como **método da bissecção**.

3.1 Objetivos

O objetivo é utilizar o método da bissecção para determinar a raiz de um polinômio $f(x)$ sobre um intervalo $[l, r]$.

3.2 Entrada

A primeira linha da entrada possui um inteiro n ($0 \leq n \leq 6$) indicando o grau do polinômio.

A segunda linha da entrada possui $n + 1$ números reais $a_0, \ldots, a_n$ ($-10 \leq a_i \leq 10$), separados por espaço, que descrevem os coeficientes do polinômio

$$f(x) = a_0 + a_1 x + a_2 x^2 + \ldots + a_n x^n$$

A terceira linha da entrada contém dois números reais, l e r ($-10^{10} \leq l \leq r \leq 10^{10}$), indicando o intervalo $[l, r]$ em que a raiz deve ser buscada.

3.3 Saída

Deverá ser impresso uma única linha contendo uma raiz de $f(x)$ sobre o intervalo $[l, r]$. Em caso de mais de uma raiz, qualquer uma poderá ser impressa. Um número x será considerando raiz desde que $|f(x)| \leq 10^{-3}$.

3.4 Exemplos

Entrada

```
3
1 -2 -8 1
1
-5.2 0.1
```

Saída

```
-0.480973
```

Entrada

```
2
-1 -2 1
-10.0 10.0
```

Saída

```
2.414551
```

3.5 Instruções de Entrega

Um arquivo chamado `bisseccao.c` deverá ser criado contendo a implementação acima e deve estar anexo a pasta a ser enviada via o formulário de entrega.

4 Critérios de Correção

A Tabela 1 especifica o peso de cada tarefa de implementação. Será descontado pontos caso o código esteja mal formatado e/ou não apresente documentação na forma de comentários.

Tabela 1: Critérios de correção.

Implementação	Peso
Épsilon da Máquina	20
Séries de Taylor	40
Método da bissecção	40
Total	100