

Instituto de Computação da UNICAMP

Disciplina MC102: Algoritmos e Programação de Computadores - Turmas W Y

Laboratório em Sala 07

Prazo de entrega: **29/06/2016 às 21:00:00**

PROFESSOR: Lehilton Lelis Chaves Pedrosa

Seno e Cosseno

Neste laboratório vamos exercitar o uso de funções.

Vamos organizar uma calculadora que calcula as funções seno e cosseno usando sequências, não vamos usar a biblioteca `math.h`. O seu programa deve receber uma ordem entre as seguintes possibilidades e calcular o que for pedido:

- Comando Seno (indicado pela letra 'S'): deve-se calcular a função seno em um dado ponto. Esta aproximação deve ser feita usando a série de Maclaurin (Série de Taylor no ponto 0). Deverá ser lido o ponto 'x', do tipo **double**, em que a função deve ser calculada e o índice 'n' que limita até onde a série deve ser calculada. Para simplificar vamos definir a série como a figura abaixo. Com o valor calculado, você deve exibir a mensagem "O valor aproximado de seno no ponto 'x' eh 'y'.", onde 'y' é o valor calculado.

$$\text{sen}(x) = \sum_{i=0}^n \frac{(-1)^i}{(2i+1)!} x^{2i+1} = x - \frac{x^3}{3!} + \frac{x^5}{5!} - \dots$$

- Comando Cosseno (indicado pela letra 'C'): deve-se calcular a função cosseno em um dado ponto. Esta aproximação deve ser feita usando a série de Maclaurin (Série de Taylor no ponto 0). Deverá ser lido o ponto 'x', do tipo **double**, em que a função deve ser calculada e o índice 'n' que limita até onde a série deve ser calculada. Para simplificar vamos definir a série como a figura abaixo. Com o valor calculado, você deve exibir a mensagem "O valor aproximado de cosseno no ponto 'x' eh 'y'.", onde 'y' é o valor calculado.

$$\text{cos}(x) = \sum_{i=0}^n \frac{(-1)^i}{(2i)!} x^{2i} = 1 - \frac{x^2}{2!} + \frac{x^4}{4!} - \dots$$

Cada um destes cálculos devem ser implementados em uma função separada que será descrita a seguir. A sua calculadora deve funcionar, recebendo comandos e fazendo cálculos, até que o usuário entre com o comando 'f', ou 'F'. Caso o usuário entre com algum comando inválido, o programa deve avisar "Comando invalido. Entre com outro comando, ou 'F' para terminar."

Além destes comandos, você deve também implementar uma função para calcular potências inteiras não negativas de um número qualquer e uma para calcular o fatorial de um número. Você pode implementar também qualquer outra função que julgar necessária.

Implementação

Você deve escrever as seguintes funções:

```
/*Esta funcao calcula o fatorial de um numero inteiro.*/
/*Recebe um inteiro e retorna o fatorial deste numero como um numero inteiro.*/
int fatorial(int a);

/*Esta funcao calcula a potencia inteira positiva de um numero double.*/
/*Recebe um double e um inteiro e calcula o valor do double elevado ao inteiro.*/
/*Retorna o valor do calculo em double.*/
double potencia(double x, int n);

/*Esta funcao calcula um valor aproximado para a funcao seno em um ponto proximo de 0.*/
```

```
/*Recebe um numero do tipo double e um limite de iteracoes para calcular o valor de*/  
/* seno no ponto do double fazendo n+1 (i=0 até i=n) iteracoes na série de Maclaurin.*/  
double seno(double x, int n);  
  
/*Esta funcao calcula um valor aproximado para a funcao cosseno em um ponto proximo de 0.*/  
/*Recebe um numero do tipo double e um limite de iteracoes para calcular o valor de*/  
/* cosseno no ponto do double fazendo n+1 (i=0 até i=n) iteracoes na serie de Maclaurin.*/  
double cosseno(double x, int n);
```

O seu programa deve ler da entrada padrão um dos comandos listados: 'S' para Seno, e 'C' para Cosseno. O programa deve fazer a operação exigida, escrever o resultado na tela com 4 casas decimais e esperar pelo próximo comando. O comando 'f', ou 'F' termina o programa sem nenhuma mensagem. Caso um comando inválido seja pedido, o programa deve dar uma mensagem de erro "Comando invalido. Entre com outro comando, ou 'F' para terminar." e esperar pelo próximo comando.

Dicas

Lembre-se de testar todas as funções separadamente enquanto escreve o programa. Comece pelas mais fáceis e menos dependentes. Por exemplo, a função para calcular seno usa fatorial, então faça primeiro a função que calcula fatorial, e **teste** esta função. Isto vai evitar erros e economizar o seu tempo.

Ao usar a função `scanf` para ler um caractere (com a máscara `%c`), deve-se acrescentar um espaço antes de `%c` para garantir que somente um caracter válido seja lido (ou seja, para ignorar espaços e quebras de linhas).

Observação Importante: não é permitido o uso da biblioteca `math.h`, só a biblioteca `stdio.h` poderá ser usada.

Entrada

O programa lê, da entrada padrão, um caractere como comando e se o comando for válido um número real representando onde deve-se calcular a função e um número inteiro N representando o número de iterações.

Saída

O programa podem imprimir as seguintes mensagens:

"O valor aproximado de seno no ponto <x> eh <y>." para o comando 'S', o tipo **double** <x> e o inteiro <i> como entrada.

"O valor aproximado de cosseno no ponto <x> eh <y>." para o comando 'C', o tipo **double** <x> e o inteiro <i> como entrada.

"Comando invalido. Entre com outro comando, ou 'F' para terminar." quando o comando for inválido.

Exemplos de execução

Exemplo 1

```
C  
3.14159  
5  
0 valor aproximado de cosseno no ponto 3.1416 eh -1.0018.  
S  
3.14159  
5  
0 valor aproximado de seno no ponto 3.1416 eh -0.0004.  
F
```

Exemplo 2

```
p  
Comando invalido. Entre com outro comando, ou 'F' para terminar.  
a  
Comando invalido. Entre com outro comando, ou 'F' para terminar.  
f
```

Nota: Textos em azul denotam dados de entrada, isto é, a serem lidos pelo programa.

Textos em **vermelho** denotam dados de saída, ou seja, a serem impressos pelo programa.

Observações

- Não é permitido o uso da biblioteca `math.h`, a única biblioteca permitida é a `stdio.h`;
- O número máximo de submissões é 15;
- Para a realização dos testes do SuSy, a compilação se dará da seguinte forma: `gcc -std=c99 -pedantic -Wall -o labSala07 labSala07.c`;
- Esta atividade foi baseada num laboratório de MC102 coordenado de 2015;
- N nunca é muito grande, de modo que os fatoriais necessários sempre cabem em um inteiro.

[Voltar](#)