

Segundo Exercício Programa

MAC0115 - Física Diurno 2010

Prof Roberto Marcondes Cesar Jr
Entrega até 15/10/2010

Objetivos

Os objetivos deste exercício programa são três:

1. Implementar de maneira eficiente as funções $\sin(x)$ e $\cos(x)$ por meio de aproximação numérica e testá-las com funções já existentes na biblioteca C `math.h`.
2. Cálculo numérico do trabalho τ realizado sobre uma partícula que se desloca sobre um caminho $\gamma = (x(t), y(t))$ por um campo de forças dado por $\mathbf{F}(x, y) = P(x, y)\mathbf{i} + Q(x, y)\mathbf{j}$ ¹.
3. Implementar um menu simples que permita ao usuário escolher o que o programa irá fazer.

1 Parte I

1.1 Implementação das funções $\sin(x)$ e $\cos(x)$

Você deverá implementar funções em C que calculem o valor das funções matemáticas $\sin(x)$ e $\cos(x)$, utilizando expansões em Séries de Taylor. As expansões em Série de Taylor de cada uma dessas funções são dadas:

$$\sin(x) = \sum_{k=0}^n \frac{(-1)^k}{(2k+1)!} x^{2k+1} \quad (1.1)$$

$$\cos(x) = \sum_{k=0}^n \frac{(-1)^k}{(2k)!} x^{2k} \quad (1.2)$$

As funções a serem implementadas devem receber um parâmetro x e devem devolver o valor esperado com uma precisão ϵ pré-definida no programa². Os protótipos de tais funções devem ser:

```
float seno (float x);  
float coseno (float x);
```

O cálculo de tais valores deve ser feito de maneira eficiente³. Estude as características de cada função que será implementada e de sua respectiva expansão em Taylor. Tente implementar o cálculo com o menor número de operações possível.

¹Note que \mathbf{r} é um vetor, pois está em negrito.

²Leia a seção 4.

³Isso será avaliado.

1.2 Testes automatizados

Você deverá escrever também uma função com o protótipo:

```
void teste_seno_cosseno ();
```

, que compara as funções `seno(x)` e `cosseno(x)` que você implementou com as `sin(x)` e `cos(x)` existentes da biblioteca C `math.h`⁴, e imprime na tela os resultados que foram negativos.

Note que é errado utilizar o operador `==` para fazer esta comparação, pois quase sempre haverá diferenças nas últimas casas decimais devido à precisão utilizada. Uma solução para isso é implementar uma função `int igual (float a, float b)` que devolve 1 se a diferença entre os valores a e b for menor que a precisão do programa e 0 caso contrário.

2 Cálculo de τ - Integral de Linha

2.1 Interlúdio Teórico

Considere um campo de forças $\mathbf{F}(x, y) = P(x, y)\mathbf{i} + Q(x, y)\mathbf{j}$, no qual uma partícula se move descrevendo uma trajetória dada pela curva paramétrica $\gamma(t) = (x(t), y(t))$, $t \in [a, b]$. Queremos calcular o trabalho realizado sobre a partícula pela força \mathbf{F} entre dois pontos da curva, digamos de $A = \gamma(a)$ a $B = \gamma(b)$ ⁵.

Quando \mathbf{F} é constante e γ é um segmento de reta, o trabalho é dado simplesmente pelo produto escalar $\tau = \mathbf{F} \cdot \mathbf{AB}$ ⁶. No caso geral, o trabalho é dado por $\tau = \int_a^b \mathbf{F}(\mathbf{r}) d\mathbf{r} = \int_a^b (P(x, y) \frac{dx(t)}{dt} dt + Q(x, y) \frac{dy(t)}{dt} dt)$.

Nem sempre o trabalho τ pode ser calculado analiticamente. Nestes casos, podemos obter aproximações numéricas para τ , dividindo o intervalo $[a, b]$ em subintervalos menores $a = t_0 < t_1 < \dots < t_i < \dots < t_n = b$, para então substituir cada arco $\gamma([t_i, t_{i+1}]) = \gamma_i$ por segmentos de reta $[A_i, A_{i+1}]$, onde $A_i = \gamma(t_i) = (x(t_i), y(t_i))$, com $i = 0, 1, \dots, n-1$.

Se cada $[A_i, A_{i+1}]$ for suficientemente pequeno, podemos supor \mathbf{F} constante em $]A_i, A_{i+1}[$ e podemos aproximar o trabalho τ_i no arco \mathbf{r}_i por:

$$\begin{aligned}\tau_i &\cong \mathbf{F}(x(t_i), y(t_i)) \cdot A_i, A_{i+1} \\ \tau_i &\cong P(x(t_i), y(t_i))\Delta x(t_i) + Q(x(t_i), y(t_i))\Delta y(t_i)\end{aligned}$$

, onde $\Delta x(t_i) = x(t_{i+1}) - x(t_i)$ e $\Delta y(t_i) = y(t_{i+1}) - y(t_i)$.

Logo, o trabalho total realizado pelo campo de forças no intervalo $[a, b]$ é:

$$\tau \cong \sum_{i=0}^{n-1} \tau_i \quad (2.1)$$

, onde n é o número de subdivisões. Note que quanto maior o número n , melhor será a aproximação.

⁴Para utilizar esta biblioteca, basta apenas colocar `#include<math.h>`.

⁵As notações seguem *Hellmeister, Ana Catarina Pontone. EDUSP, Edição 1 (1996) Cálculo Integral Avançado ISBN 8531403707*

⁶Que é o trabalho conhecido por você desde o colegio.

2.2 Implementação

O seu programa deverá:

1. Receber:

- (a) Os valores a e b , que representam os limites do intervalo de t , ou seja, o início e o fim do caminho escolhido.

2. Devolver:

- (a) O menor número de subdivisões necessário para que o erro do resultado numérico em relação ao analítico seja menor do que ϵ .⁷
- (b) O valor calculado numericamente do trabalho realizado pela força ao longo do caminho.

Para a implementação das forças, você deverá escrever também as seguintes funções:

```
float P (float x, float y, int m);  
float Q (float x, float y, int m);
```

, que devolverão respectivamente o valor de $P_m(x, y)$ e $Q_m(x, y)$ da força $\mathbf{F}_m(x, y) = P_m(x, y)\mathbf{i} + Q_m(x, y)\mathbf{j}$. O parâmetro m identifica a função selecionada.

E para a implementação dos caminhos:

```
float gamma_x (float t, int m);  
float gamma_y (float t, int m);
```

, que devolverão respectivamente o valor de $x_m(t)$ e $y_m(t)$ do caminho $\gamma_m(t) = (x_m(t), y_m(t))$.

A força e o caminho que serão utilizados, são escolhidos via menu (seção 3). As Forças serão:

$$\mathbf{F}_1(x, y) = x\mathbf{i} + y\mathbf{j} \quad (2.2)$$

$$\mathbf{F}_2(x, y) = xy\mathbf{i} + \mathbf{j} \quad (2.3)$$

$$\mathbf{F}_3(x, y) = \frac{-y}{x^2 + y^2}\mathbf{i} + \frac{x}{x^2 + y^2}\mathbf{j} \quad (2.4)$$

$$(2.5)$$

, e os caminhos (onde for possível, utilize suas implementações das funções `seno(x)` e `cos seno(x)` da seção 1.1):

$$\gamma_1 = (\cos(t), \sin(t)) \quad (2.6)$$

$$\gamma_2 = (3t^2, 2t^3) \quad (2.7)$$

$$\gamma_3 = (A\cos(t), A\sin(t)) \quad (2.8)$$

$$(2.9)$$

A curva \mathbf{r}_3 é uma circunferência com raio A , um parâmetro solicitado, caso o usuário escolha esta opção.

Para calcular o valor analítico das integrais, escreva a seguinte função:

```
float analítico (float a, float b, int m);
```

⁷**Dica:** você pode utilizar dois laços encaixados, o externo considera o erro e o número de subdivisões e o interno calcula o trabalho.

Campo	Curva	Valor Analítico
F_1	γ_1	0
F_2	γ_2	$\frac{36}{7}(b^7 - a^7) + 2(b^3 - a^3)$
F_3	γ_3	$b - a$

Tabela 1: Valores do trabalho calculados analiticamente.

3 Menu

A função do menu é mostrar ao usuário o que ele pode fazer no programa imprimindo na tela as opções. Ele deve ter a seguinte forma:

```
*****
Digite a opcao desejada:
1 - Rodar testes para as funcoes seno e cosseno
2 - Calcular integrais numéricas
*****
```

Caso seja escolhida a primeira opção, o programa deverá rodar os testes (seção 1.2) e imprimir na tela **apenas** os testes que foram negativos e o número de testes que foram negativos.

Agora, caso a segunda opção seja escolhida, aparecerá na tela o próximo menu:

```
*****
Digite uma das opcoes abaixo para escolher a forza e
o caminho:
1 - F1(x,y) = (x, y)
   gamma1(t) = (cos(t), sen (t))
2 - F2(x,y) = (x * y, 1)
   gamma2(t) = (3 * t^2, 2 * t^3)
3 - F3(x,y) = (-y/(x^2 + y^2) , x/(x^2 + y^2))
   gamma3(t) = (A * cos(t), A * sen (t))
*****
```

Você poderá implementar mais forças e mais caminhos, porém as três forças e caminhos pedidos são obrigatórios.

4 Precisão

No seu programa existem duas precisões definidas⁸:

1. A primeira é a utilizada para implentar `seno(x)` e `cosseno(x)`. Esta deve ser igual à 10^{-4}
2. A segunda é utilizada para calcular o número de subdivisões n mínimo para que a integral numérica seja "igual" à analítica. Esta deve ser 10^{-3} .

⁸Utilizem `#define`

5 Instruções

5.1 Sobre a Elaboração

O EP pode ser elaborado por equipes de dois alunos, desde que as seguintes regras sejam respeitadas:

- Os alunos devem trabalhar sempre juntos. A ideia é que deve existir uma cooperação.
- Caso em um grupo exista um aluno com maior facilidade, este deve explicar as decisões tomadas, e o seu par deve participar e se esforçar para entender o desenvolvimento do programa. (denominamos isso de *programação em pares*, que é uma excelente prática que vocês devem se esforçar para adotar).
- Mesmo a digitação do EP deve ser feita em grupo, enquanto um digita, o outro fica acompanhando o trabalho.
- Recomendamos fortemente que o exercício seja desenvolvido da forma descrita nos itens acima, mas ele também pode ser feito individualmente.

5.2 Sobre a Avaliação

- É sua responsabilidade manter o código do seu EP em sigilo, ou seja, apenas você e seu par podem ter acesso ao código.
- No caso de EPs feitos em dupla, a mesma nota será atribuída aos dois alunos do grupo.
- **Não serão toleradas cópias!** Exercícios copiados (com eventuais disfarces) levarão à reprovação da disciplina e o encaminhamento do caso para a Comissão de Graduação.
- Exercícios atrasados não serão aceitos.
- Exercícios com erro de sintaxe (ou seja, erros de compilação) receberão nota ZERO.
- É muito importante que seu programa tenha comentários e esteja bem identado, ou seja, digitado de maneira a ressaltar a estrutura de subordinação dos comandos do programa. Isto irá influenciar a sua nota.
- Caso o programa apresente resultados "estranhos" (inesperados) para eventuais dados de entrada "incorretos", haverá desconto de nota.
- As informações impressas pelo seu programa na tela devem aparecer da forma mais clara possível. Este aspecto também será avaliado.
- Uma regra básica é a seguinte: do ponto de vista do monitor responsável pela correção dos trabalhos, quanto mais convenientemente apresentado estiver o seu programa, melhor avaliado ele será.

5.3 Sobre a entrega

- O prazo de entrega é até o dia 15/10/2010.
- Caso feito em dupla: **Ambos devem submeter a versão final do código!**
- Entregar apenas um arquivo com o nome CalcNum.c.

- Para a entrega, utilize o Paca. Você pode entregar várias versões de um mesmo EP até o término do prazo, mas somente a última versão que permanecerá armazenada pelo sistema.
- Não serão aceitas submissões por e-mail ou atrasadas. Não deixe para a última hora, pois o sistema pode ficar congestionado, e você corre o risco de não conseguir enviar.
- Guarde uma cópia do seu EP pelo menos até o final do semestre.
- No início do arquivo, acrescente o seguinte cabeçalho:

```

/*****/
/** MAC0115 IFUSP Diurno 2010                               **/
/** Prof Roberto Cesar                                     **/
/**                                                         **/
/** Segundo Exercicio Programa -- Calculo Numerico         **/
/**                                                         **/
/** <nome do(a) aluno(a)> <numero USP>                     **/
/** <nome do(a) aluno(a)> <numero USP>                     **/
/**                                                         **/
/** Informacoes sobre Desenvolvimento:                     **/
/** <Ambiente (Dev-C++, CodeBlocks, Gedit)>                 **/
/** <Sistema Operacional (Windows, Linux)>                 **/
/*****/

```