

MAC 110 – Introdução à Ciência da Computação  
Prof. Marcel Parolin Jackowski  
Bacharelado em Ciência da Computação  
IME/USP – PRIMEIRO SEMESTRE DE 2012  
**Segundo Exercício-Programa**  
Data de entrega: até 24/maio/2012

## Introdução

Este exercício-programa terá duas partes: (1) o cálculo aproximado de funções matemáticas, tais como funções trigonométricas, através de operações matemáticas básicas (+, -, \* e /) e (2) a medição do desempenho do cálculo das funções matemáticas.

### Primeira Parte – Funções matemáticas

Você deverá escrever uma classe contendo métodos para calcular as funções matemáticas seno, cosseno, raiz quadrada e  $\ln$  (logaritmo na base  $e$ ). Para tanto, use a fórmula dada pela série de Taylor:

- $\text{sen}(x) = \frac{x}{1!} - \frac{x^3}{3!} + \frac{x^5}{5!} - \dots + \frac{(-1)^k x^{2k+1}}{(2k+1)!}, x \in R$
- $\text{cos}(x) = 1 - \frac{x^2}{2!} + \frac{x^4}{4!} - \frac{x^6}{6!} + \dots + \frac{(-1)^k x^{2k}}{(2k)!}, x \in R$
- $\ln(1+x) = x - \frac{x^2}{2!} + \frac{x^3}{3!} - \frac{x^4}{4!} + \dots + \frac{(-1)^{(k-1)} x^k}{k}, |x| < 1$
- $\sqrt{1+x} = 1 + \frac{1}{2}x - \frac{1}{8}x^2 + \frac{1}{16}x^3 - \frac{5}{128}x^4 + \dots + \frac{(-1)^k (2k)!}{(1-2k)(k!)^2 (4^k)} x^k, |x| < 1$
- Outras fórmulas podem ser encontradas em bons livros de cálculo ou em [http://en.wikipedia.org/wiki/Taylor\\_series](http://en.wikipedia.org/wiki/Taylor_series).

Procure implementar estas funções de uma forma ao mesmo tempo clara (que seja fácil de entender) e eficiente (que seja rápida, ou seja, que execute poucas operações). Sinta-se à vontade para utilizar métodos (e, talvez, classes) auxiliares se isso for ajudar na clareza. O cálculo de  $\ln$  possui uma pequena complicação. Note que a fórmula dada tem  $1+x$  como parâmetro e não  $x$ . Assim, para implementá-la, deve-se criar um método `double ln(double umMaisX)` e no seu interior definir uma variável local `x = umMaisX - 1` de modo que se possa calcular  $\ln(1+x)$ . Veja que o cálculo da raiz quadrada deverá seguir o mesmo raciocínio.

A classe que contém as funções matemáticas deverá conter também um método através do qual definir-se-á a precisão do cálculo. Se este método não for chamado, a precisão padrão a ser adotada será  $1.0e-8$ , ou seja, o erro máximo permitido no cálculo aproximado será de 8 casas decimais. Usuários que desejarem uma precisão maior ou menor poderão usar este método para alterá-la.

### Segunda Parte – Teste de desempenho

Agora, para terminar, vamos avaliar o desempenho da nossa implementação, medindo quanto tempo ele demora para calcular cada uma das funções matemáticas implementadas. O cálculo do tempo pode ser feito através da utilização do método `System.nanoTime()`, que devolve o número de nanosegundos do relógio interno do computador. Para saber o quanto tempo demora a execução de um método, basta chamar `System.nanoTime()` antes e depois do método e subtrair um tempo do outro.

Você deverá escrever uma classe medidora de desempenho que imprimirá, de uma forma clara, precisa e fácil de ler, um relatório sobre o desempenho das funções. Este relatório deve mostrar o tempo médio (em milissegundos) e o respectivo desvio padrão de 100 execuções de cada uma das funções matemáticas.

Finalmente, para a função seno, você deve ainda escrever um método que calcula (e imprime) qual o ganho de desempenho (termos percentuais) ao se usar o tipo float ao invés do tipo double para realizar todos os cálculos do seno.

## Observações finais

### Sobre a elaboração:

Este EP pode ser elaborado por equipes de um ou dois alunos, desde que sejam respeitadas as seguintes regras.

- Os alunos devem trabalhar sempre juntos cooperativamente.
- 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 (chamamos 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.

### Sobre a avaliação:

- É sua responsabilidade manter o código do seu EP em sigilo, ou seja, apenas você e seu par devem ter acesso ao código.
- **Não serão toleradas cópias!** Exercícios copiados (com ou sem eventuais disfarces) levarão à reprovação da disciplina e o encaminhamento do caso para a Comissão de Graduação do aluno.
- Exercícios com erros de sintaxe (ou seja, erros de compilação) receberão nota zero.
- É muito importante que seu programa seja elegante, claro e bem indentado, ou seja, digitado de maneira a ressaltar a estrutura de subordinação dos comandos do programa (conforme visto em aula). A qualidade do seu trabalho sob esse ponto de vista influenciará sua nota!
- As informações impressas pelo seu programa na tela devem aparecer da forma mais clara possível. Este aspecto também será levado em consideração no cálculo da sua nota.
- 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 será seu trabalho.

### Sobre a entrega:

- Entregar apenas um arquivo de nome **Matematica.zip** contendo todas as classes de seu exercício. Se quiser entregar também um arquivo texto contendo uma explicação sobre o programa (opcional), dê a ele o nome **LEIAME**.
- No início de cada arquivo, acrescente um cabeçalho bem informativo, como o seguinte:

```
/*****  
/**  MAC 110 - Introdução à Computação          **/  
/**  IME-USP - Primeiro Semestre de 2012        **/  
/**  <turma> - <nome do professor>              **/  
*****/
```

```

/**                                                    **/
/** Segundo Exercício-Programa                        **/
/** Arquivo: MedidorDeDesempenho.java                **/
/**                                                    **/
/** <nome do(a) aluno(a)>                            <número USP>    **/
/** <nome do(a) aluno(a)>                            <número USP>    **/
/**                                                    **/
/** <data de entrega>                                **/
/*****/

```

Não é obrigatório que o cabeçalho seja idêntico a esse, apenas que contenha pelo menos as mesmas informações.

- Para a entrega, utilize o Paca. Você pode entregar várias versões de um mesmo EP até o prazo, mas somente a última será armazenada pelo sistema.
- Não serão aceitas submissões por email ou atrasadas. Não deixe para a última hora, pois o sistema pode ficar congestionado e você poderá não conseguir enviar.
- Guarde uma cópia do seu EP pelo menos até o fim do semestre e, novamente, você é responsável por manter o sigilo de seu código-fonte.