Departamento de Ciência da Computação – IME-USP

# MAC 115 – Introdução à Computação

Instituto de Física — Primeiro Semestre de 2005

Segundo Exercício-Programa — Data de entrega: até **12 de maio de 2005.**

## Caça-Níqueis

Uma máquina de apostas "caça-níqueis" funciona da seguinte forma: a máquina tem três rolos ($R_1, R_2$ e $R_3$), sendo que cada rolo tem cinco figuras ($f_1, f_2, f_3, f_4$ e $f_5$). Cada vez que a máquina é acionada, os rolos giram até parar em uma posição aleatória, apresentando uma das figuras.

1. Quando dois rolos apresentam figuras iguais e apenas um apresenta figura diferente, a máquina premia o apostador com uma quantia $X$.
2. Quando os três rolos apresentam figuras iguais mas essas figuras não são a figura $f_1$, a máquina premia com uma quantia $2 \times X$.
3. Quando os três rolos apresentam a figura $f_1$, o prêmio é $4 \times X$.
4. Nos casos restantes, o apostador fica a ver navios.

Escreva um programa para simular o funcionamento dessa máquina. Para tanto, voce precisará de um gerador de números aleatórios entre 1 e 5. Para implementar o gerador de números aleatórios, seu programa deve proceder da seguinte maneira:

Primeiro, o programa deve inicializar uma variável $y$ com um valor real:

```
y = 0.<seu número USP>                          /* exemplo:  y = 0.5262199 */
```

Depois deve ser feito o seguinte cálculo:

```
x = (9821.0 * fabs(seno(y)) + 0.211327) ........................................ (1)
y = x - floor(x) ............................................................... (2)
```

onde `fabs(x)` é uma função que devolve o módulo de `x`,

`floor(x)` é uma função que devolve o maior inteiro não maior que `x`,

`seno(x)` é uma função que calcula o seno de $x$ *usando o método descrito adiante.*

As fórmulas em (1) e (2) (nesta ordem) fornecem um número no intervalo $[0, 1[$. Para obter um número inteiro no intervalo [1,5] basta fazer o seguinte cálculo:

```
figura = (y * 5) + 1; ......................................................... (3)
```

Observe que a variável "figura" fornece uma figura sorteada. Para obtermos novas figuras, os cálculos (1), (2) e (3) devem ser repetidos considerando o valor mais recente de "y". Como o valor de "y" é alterado a cada novo cálculo, obtemos sempre uma nova figura como resultado.

Se a distribuição das figuras produzidas dessa maneira for uniforme, as seguintes probabilidades devem ser obtidas (experimente deduzi-las e verificar se os valores abaixo estão corretos):

$$\text{prob}(X) = \tfrac{14}{25}, \quad \text{prob}(2 \times X) = \tfrac{4}{125}, \quad \text{prob}(4 \times X) = \tfrac{1}{125}, \quad \text{prob(ver navios)} = \tfrac{2}{5}.$$

Seu programa deve permitir que o usário escolha uma entre quatro alternativas:

1. Apostar apenas na figura $f_1$: qualquer outro prêmio será desconsiderado.
2. Apostar somente em obter três figuras iguais: prêmio correspondente a apenas duas figuras iguais será desconsiderado.
3. Apostar em qualquer prêmio: o que vier será lucro.
4. Modo de aferição: nesse modo o programa deverá fazer o seguinte:
   (a) Ler e imprimir um número inteiro $N \geq 1$;
   (b) Apostar $N$ vezes e contar quantas vezes cada prêmio é gerado;
   (c) Imprimir um histograma na posição horizontal (como mostrado no exemplo abaixo) com 180 asteriscos (*), onde o número de asteriscos na linha com o **Prêmio** $i$ é proporcional ao número de ocorrências do prêmio $i$ durante as $N$ apostas;
   (d) Imprimir a diferença entre a quantidade de prêmios gerados e a quantidade teórica que deveria ter sido obtida conforme acima, para cada prêmio.

### Cálculo do seno

Seu programa deve ter pelo menos uma função:

```
double seno(double x)
{
    ...
}
```

Essa função deve calcular os valores do seno($x$) **USANDO OBRIGATORIAMENTE** a seguinte aproximação:

$$\text{seno}(x) = \frac{x}{1!} - \frac{x^3}{3!} + \frac{x^5}{5!} - \frac{x^7}{7!} + \cdots + \frac{(-1)^k x^{2k+1}}{(2k+1)!} + \cdots$$

sendo incluídos na soma todos os termos enquanto

$$\frac{|x^{2k+1}|}{(2k+1)!} \geq 10^{-8}.$$

### Exemplo de funcionamento de programa

A saída do programa não precisa ser exatamente como a mostrada a seguir. Você pode deixá-la mais bonita, se o desejar, colocando, por exemplo, o histograma na vertical (não é complicado, pense a respeito).

```
Bem vindo ao programa CaçaNíqueis!!! Suas opções são:

    0 - sair do programa
    1 - aposta só na figura f1
    2 - aposta em três figuras iguais
    3 - aposta em qualquer prêmio
    4 - modo de aferição

Digite a opção desejada: 1
Você apostou em f1...
Figuras sorteadas: f2 f4 f3
Infelizmente você perdeu!

Digite a opção desejada: 3
Você apostou em qualquer prêmio...
Figuras sorteadas: f5 f3 f5
Você ganhou X!

Digite a opção desejada: 3
Você apostou em qualquer prêmio...
Figuras sorteadas: f2 f2 f2
Você ganhou 2*X!

Digite a opção desejada: 4
Modo de aferição, digite o número de apostas: 60
Valor de N: 60
Resultados:

  Prêmio                1           2           3           4
  Ocorrências          15          14          12          19

  Histograma:
    1     *********************************************
    2     ****************************************
    3     ************************************
    4     *********************************************************

  Premio X:  esperado 33.6, obtido 15, diferenca -18.6
  Premio 2X: esperado 1.9, obtido 14, diferenca 12.1
  Prêmio 4X: esperado 0.5, obtido 12, diferenca 4.3
  Prêmio 0:  esperado 24.0, obtido 19, diferenca: -5.0

Digite a opção desejada: 0
Volte sempre!!!
```

# Observações Importantes sobre os Exercícios-Programas

Todos os exercícios-programas devem ter o seguinte cabeçalho:

```
/*******************************************************/
/* Fulano de Tal                                       */
/* Exercicio-Programa xx                               */
/* Curso yy  - Turma zz  -- Professor:  Ciclano de Tal */
/* Computador: .....                                   */
/* Compilador: .....                                   */
/*******************************************************/
```

1) O exercício-programa é estritamente individual. Exercícios copiados (com ou sem eventuais disfarces) receberão nota ZERO.
2) Exercícios atrasados não serão aceitos.
3) Exercícios com erros de sintaxe (ou seja, erros de compilação) receberão nota ZERO.
4) É muito importante que seu programa tenha comentários e esteja bem indentado, ou seja, digitado de maneira a ressaltar a estrutura de subordinação dos comandos do programa (conforme visto em aula). A avaliação dos exercícios-programas levará isto em conta.
5) Cada programa deve ser executado tantas vezes quantas forem necessárias para testar todos os casos possíveis para as entradas.