

Fundamentos de Programação

Geração de números “aleatórios”

Dainf - UTFPR

Profa. Leyza B. Dorini

Prof. Bogdan T. Nassu



Números aleatórios? Para quê?

Números aleatórios são bastante úteis em certas áreas:

- Simulações e games.
- Algoritmos probabilísticos de busca e otimização.
- Teste de software.
- Computação gráfica.

São duas as principais razões para se usar números aleatórios:

- Gerar valores e casos de teste em grandes quantidades.
- Adicionar incerteza e variação, para evitar comportamentos repetitivos/previsíveis ou gerar padrões de aspecto mais natural.

Daqui para a frente, usaremos números aleatórios com certa frequência, principalmente para gerar dados de entrada/teste.

Como gerar números aleatórios?

Usaremos funções da biblioteca-padrão de C.

Por ser uma solução algorítmica, os resultados não são de fato aleatórios: são “pseudoaleatórios”, ou “aparentemente aleatórios”.

Para nossos fins, é o suficiente: basta que os números gerados sigam uma distribuição aproximadamente uniforme, ou seja, todos os valores devem ter aproximadamente a mesma probabilidade de aparecer.

Se você precisar de números aleatórios “de verdade”, existem soluções baseadas em hardware, no estado do SO, e até na medição de fenômenos naturais.

Números aleatórios em C

A biblioteca-padrão `stdlib` tem as funções que precisamos:

```
int rand ();
```

Retorna um valor entre 0 e `RAND_MAX`, constante definida no arquivo `stdlib.h`.

```
void srand (unsigned int seed);
```

Inicializa a “semente” dos números aleatórios. A sequência gerada pela função `rand` é sempre a mesma para uma mesma semente aleatória.

Alguns truques úteis

Para gerar sequências diferentes a cada execução, inicialize a semente aleatória com um valor diferente a cada execução. Podemos fazer isso usando a função `time`, declarada em `time.h`:

```
srand (time (NULL));
```

O que acontece se o programa for executado duas vezes durante o mesmo segundo?

Para limitar a faixa de valores, podemos usar o operador `%`:

```
x = rand () % 10; → Qualquer valor de x ficará entre 0 e 9!
```

Para gerar valores com casas decimais, podemos dividir o número por `RAND_MAX`, obtendo um valor entre 0 e 1, e multiplicar o valor obtido por algum valor máximo menor que `RAND_MAX`.