

Programación en C++

Números Aleatorios

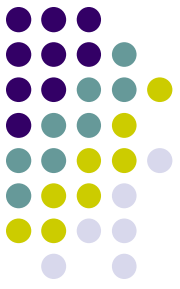




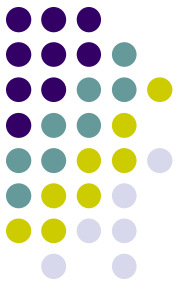
Números Aleatorios

- Los números aleatorios son con frecuencia muy importantes en programación
 - Suponga que escribirá un programa para jugar el juego de la ruleta
 - Los números que aparecen en una ruleta real son aleatorios
 - Escribir un programa con valores predecibles no es un juego de ruleta muy interesante
 - Queremos simular aleatoriedad en el computador!
 - ¿Qué pasa con las máquinas tragamonedas computarizadas?

Computadoras y Números Aleatorios



- Las computadoras son máquinas “determinísticas”.
- Las computadoras no son capaces de una verdadera aleatoriedad
- En su lugar, se utilizan números pseudo-aleatorios
 - Los números pseudo-aleatorios se generan usando una formula matemática.
 - Hay un patrón en números pseudo-aleatorios
 - El patrón es casi imposible de observar



Ventaja de Pseudo-Aleatoriedad

- Sería difícil depurar programas que usen números verdaderamente aleatorios.
- Al proporcionar el mismo valor de inicio (semilla) al generador, se pueden observar los mismos resultados de secuencia en cada ejecución.
- Cuando haya terminado la depuración, puede usar una semilla diferente cada vez.
- A menudo, esto se hace usando el reloj del sistema..



Números Aleatorios en C++

- Se debe incluir el archivo de cabecera `<cstdlib>`
- Funciones proporcionadas:
 - `void srand(unsigned int seed);`
 - Esta funcion fija la semilla para el generador de números aleatorios al valor del parámetro semilla
 - `int rand();`
 - Esta funcion genera y retorna un valor entero en el rango `0..RAND_MAX` (que está definido como constante en `cstdlib`)



Números Aleatorios en C++

- `srand()` es llamado usualmente solo UNA vez
- `rand()` es llamado cada vez que se desea generar un número aleatorio.
 - Si deseamos obtener un número aleatorio entre 0 y N:
 - $\text{val} = \text{rand()} \% (N+1)$
 - Si deseamos obtener un número aleatorio en el rango M..N
 - $\text{val} = \text{rand()} \% (N-M+1) + M$

```
#include <cstdlib> //req para srand y rand
#include <iostream> //req para cout y cin
using namespace std;
```

```
int main()
{
    double avg = 0.0;
    int i, minX = 30, maxX = 50;
    int randVal, sem;
    cout << "Ingrese semilla: ";
    cin >> sem;
    srand(sem);
    for (i = 0; i < 10; i++)
    {
        randVal = rand() % (maxX - minX + 1) + minX;
        cout << randVal << endl;
    }
    for (i = 0; i < 10000; i++)
    {
        avg += rand() % (maxX - minX + 1) + minX;
    }
    avg = avg / 10000;
    cout << "Media de 10000: " << avg << endl;
    return (0);
}
```

srand() es llamado usualmente solo una vez al iniciar una secuencia

rand() es llamada cada vez que se necesite un número pseudo aleatorio



Semilla: 12

42

46

40

41

40

43

32

38

31

42

Media de 10000: 39.9971

Semilla: 1652

38

32

43

43

36

48

34

48

31

49

Media de 10000: 40.0484

Semilla: 12

42

46

40

41

40

43

32

38

31

42

Media de 10000: 39.9971

Nota: Igual semilla = igual secuencia = iguales resultados


```
//include iostream, cstdlib, and cmath
int main()
{
    long int stopX;
    srand(100);
    stopX = (long int)pow(2, 32) - 1;
    long int li;

    cout << "RANDOM #" << rand() << endl;
    cout << "RANDOM #" << rand() << endl;
    for (li = 0; li < stopX; li++)
        rand();
    cout << "RANDOM #" << rand() << endl;
    cout << "RANDOM #" << rand() << endl;
    for (li = 0; li < stopX; li++)
        rand();
    cout << "RANDOM #" << rand() << endl;
    cout << "RANDOM #" << rand() << endl;
    return (0);
}
```

RANDOM #12662
RANDOM #23392
RANDOM #12662
RANDOM #23392
RANDOM #12662
RANDOM #23392

While the pattern **DOES** repeat, there are about *4.3 billion* "random" numbers generated in the pattern.
(This program took about 29 minutes of time to run)

Uso de Time como Semilla del Generador



- Debe incluir la biblioteca `<time.h>`
 - Proporciona la función “`time()`”
- Llamada a la función “`time()`”
 - `time(NULL)`
 - Retorna el tiempo actual del reloj del sistema
 - Expresado en número de segundos transcurridos desde 00:00 horas, Ene 1, 1970 UTC

```

#include <cstdlib> //req para srand y rand
#include <iostream> //req para cout and cin
#include <time.h> //req para usar time
using namespace std;

int main()
{
    double avg = 0.0;
    int i, minX = 30, maxX = 50;
    int randVal, seed;
    srand( time(NULL) );

    for (i = 0; i < 10; i++)
    {
        randVal = rand() % (maxX - minX + 1) + minX;
        cout << randVal << endl;
    }
    for (i = 0; i < 10000; i++)
    {
        avg += rand() % (maxX - minX + 1) + minX;
    }
    avg = avg / 10000;
    cout << "Media de 10000: " << avg << endl;
    return (0);
}

```

srand() es llamado usualmente Solo una vez al iniciar una secuencia

Usar el reloj para inicializar el generador en vez de la entrada del usuario

rand() es llamada cada vez que se necesite un número pseudo aleatorio