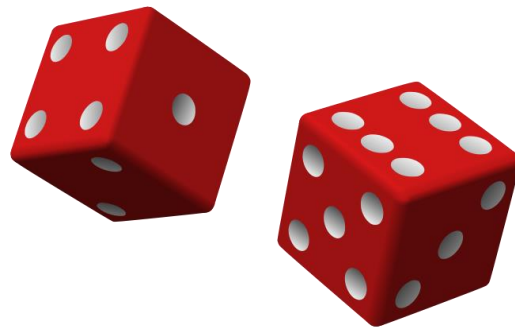


COMO SE GENERAN NÚMEROS ALEATORIOS EN C++ EJEMPLOS

Primero debes saber que:

- Los números aleatorios son de gran importancia en la programación. Necesitas números aleatorios para hacer girar unos dados, repartir cartas, posicionar enemigos en la pantalla y tal vez hasta controlar sus movimientos.
-



Para generar un número aleatorio en C++ se utiliza la función predefinida llamada **rand** cuya sintaxis de uso es la siguiente:

```
rand()%(N+1)
```

(En caso no se haya configurado el estándar ISO C++11 se deberá incluir la librería **cstdlib**)

Valor entero positivo

La función rand retorna un valor aleatorio entero en el intervalo de [0 , N]

Para que la función **rand** puede **tender** a generar números aleatorios de valor distinto cada vez que es invocada, se debe utilizar **previamente** la siguiente instrucción:

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

(se debe incluir la librería **ctime** para usar **time**)

(En caso no se haya configurado el estándar ISO C++11 se deberá incluir la librería **cstdlib** para usar **srand**)

Esta instrucción activa la “semilla” de números aleatorios

Hace que la computadora lea su reloj para obtener automáticamente un valor aleatorio distinto en el mismo día.



Ejemplo

- Realizar un programa que permita generar un valor aleatorio entero en el rango $[0, 10]$, y mostrarlo en pantalla.

Solución:

```
#include <iostream>
#include <ctime>

using namespace std;

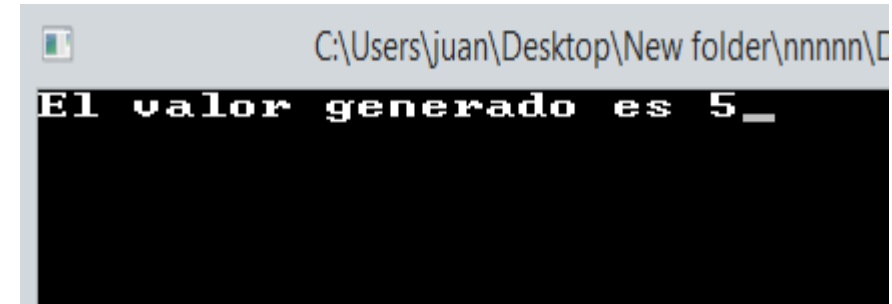
int main()
{
    int x;

    srand(time(NULL)); //activa la semilla de números aleatorios
    x = rand() % (10 + 1); //se genera un aleatorio en el rango [0,10]
    cout<<"El valor generado es: "<<x; //se muestra el valor del numero
    return 0;
}
```

Se deben incluir las siguientes librerías:

`<ctime>` //para usar la función `time`
`<cstdlib>` //para usar las funciones `srand` y `rand`
//en caso se necesite

Resultado de ejecutar el programa



Si el programa se ejecuta mas de una vez los valores aleatorios **tenderán** a salir distintos debido a que previamente al uso de **rand()** se ha incluido la instrucción **srand(time(NULL))**

Se ha aplicado la sintaxis : **rand()%(N+1)** , también se pudo haber colocado de frente **rand()%(11)**

En el ejemplo anterior, también se puede haber mostrado el resultado directamente el comando `cout` sin necesidad de usar una variable adicional.

```
srand(time(NULL)); //activa la semilla de números aleatorios  
cout<<"El valor generado es:"<<rand() % (10 + 1); //se genera el número aleatorio  
//en el rango [0,10] y se muestra  
//en pantalla.
```

Si se desea generar un numero aleatorio en el rango **[a , b]** (para cualquier valor entero de a y b) se deberá utilizar la siguiente regla:

$$\text{rand()} \% (b - a + 1) + a$$

Esta regla se cumple siempre y cuando $b > a$

Recuerde:

La instrucción ***srand(time(NULL))*** debe colocarse **ANTES** de invocar a la function ***rand***.





Ejemplo

- Realizar un programa que permita simular el valor obtenido de lanzar un dado y mostrar dicho valor en pantalla.

Solución:

```
#include <iostream>
#include <ctime>

using namespace std;

int main()
{
    int d;

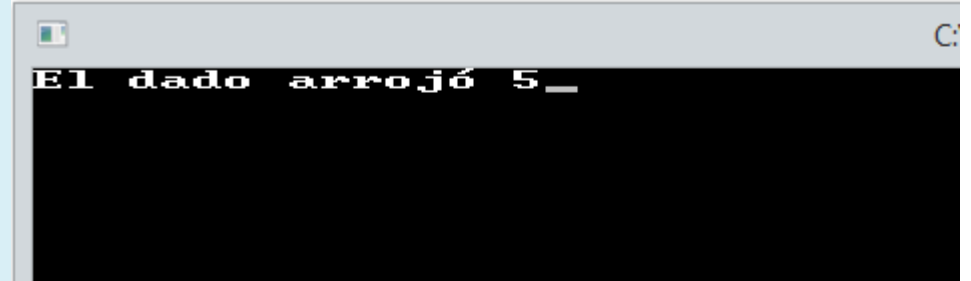
    srand(time(NULL)); //activa la semilla de números aleatorios

    d = rand() % (6-1 + 1)+1; //se genera un aleatorio en el rango [1,6]

    cout<<"El dado arrojó: "<<d; //se muestra el resultado

    return 0;
}
```

Resultado de ejecutar el programa



El dado arrojó 5_

Si el “**dado se vuelve a lanzar**” los resultados tenderán a salir distintos debido a que previamente al uso de **rand()** se ha incluido la instrucción **srand(time(NULL))**

Se ha aplicado la regla : $\text{rand()} \% (b-a+1) + a$, también se pudo haber colocado defrente $\text{rand()} \% (6) + 1$

En el ejemplo anterior, también se puede haber mostrado el resultado directamente con la función `printf`, sin necesidad de usar una variable adicional.

[illegible]

```
cout<< "El dado arrojó: " << rand() % (6-1+1)+1, //se genera el número aleatorio
//en el rango [1,6] y se muestra
//en pantalla.
```



Ejemplo

Ejecuta el siguiente programa mas de una vez, y analiza el resultado. **¿Qué crees que está pasando?**

```
#include <iostream>
#include <ctime>

using namespace std;

int main()
{
    cout<<"El valor generado es: "<<rand()%(10+1);//se muestra el resultado

    return 0;
}
```

Respuesta:

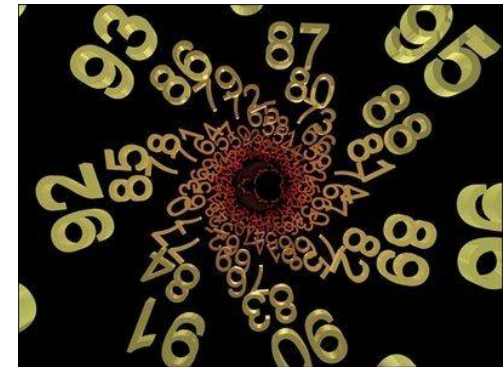
Debido a que no se ha colocado previamente la instrucción:

srand(time(NULL))

Todos los valores generados serán iguales al primero que se generó

Es importante recalcar que la generación computacional de números aleatorios en el sentido estricto de la palabra es casi imposible, lo que hace **rand** es generarlos a través de reglas matemáticas determinísticas.

*Por lo que a dichos números generados se les llama **pseudoaleatorios***





Importante

- ➔ *La función rand retorna números aleatorios de tipo entero.*
- ➔ *Se pueden generar números aleatorios en un rango entero utilizando una regla muy simple.*
- ➔ *Si no se coloca la sentencia srand(time(NULL)) antes de la función rand el número aleatorio generado siempre será el mismo para para cada ejecución realizada*
- ➔ *La función rand genera en realidad números p-seudoaleatorios.*