

Generación de números aleatorios

El procedimiento que realicé para obtener números aleatorios fue el de obtener datos del hardware de mi PC mediante la función **GetSystemInfo** de la librería **windows.h**, además busqué la forma de obtener el tiempo que tarda en obtener estos datos y mostrarlos en nanosegundos. Luego de esto convierto tanto los datos constantes que son los datos de mi PC y el dato “aleatorio” que es el tiempo de ejecución a NTL y los multiplico para luego añadirlos a un array que almacenará estos números aleatorios.

```
{  
    //Iniciar el timer  
  
    auto start = std::chrono::high_resolution_clock::now();
```

- Elijo ocho datos entre los que puede retornar la función **GetSystemInfo**.

```
//Obtener información del hardware de la PC  
  
printf("Hardware info: \n");  
  
std::cout << std::endl;  
  
printf("  OEM ID: %u\n",  
       siSysInfo.dwOemId);  
printf("  Number of processors: %u\n",  
       siSysInfo.dwNumberOfProcessors);  
printf("  Page size: %u\n",  
       siSysInfo.dwPageSize);  
printf("  Processor type: %u\n",  
       siSysInfo.dwProcessorType);  
printf("  Active processor mask: %u\n",  
       siSysInfo.dwActiveProcessorMask);  
printf("  Processor level: %u\n",  
       siSysInfo.wProcessorLevel);  
printf("  Processor Architecture: %u\n",  
       siSysInfo.wProcessorArchitecture);  
printf("  Processor Revisison: %u\n",  
       siSysInfo.wProcessorRevision);
```

- Detengo el timer y convierto todos los datos obtenidos a NTL.

```
//Convertir los datos a NTL
ZZ a(siSysInfo.dwOemId);
ZZ b(siSysInfo.dwNumberOfProcessors);
ZZ c(siSysInfo.dwPageSize);
ZZ d(siSysInfo.dwProcessorType);
ZZ e(siSysInfo.dwActiveProcessorMask);
ZZ f(siSysInfo.wProcessorLevel);
ZZ g(siSysInfo.wProcessorArchitecture);
ZZ h(siSysInfo.wProcessorRevision);

//Detener el timer
auto finish = std::chrono::high_resolution_clock::now();

//Mostrar el tiempo obtenido en nanosegundos
std::cout << "\nTime: " << std::chrono::duration_cast<std::chrono::nanoseconds>(finish-start).count() << " nanoseconds\n";

//Convertir el tiempo obtenido a NTL
ZZ t(std::chrono::duration_cast<std::chrono::nanoseconds>(finish-start).count());
```

- Finalmente multiplico los datos de la PC por el tiempo obtenido y los añado a un array.

```
ZZ t(std::chrono::duration_cast<std::chrono::nanoseconds>(finish-start).count());

//Multiplicar los datos de la PC por el tiempo y añadirlos a un array
ZZ Array[8] = {a * t, b * t, c * t, d * t, e * t, f * t, g * t, h * t};

std::cout << std::endl;

//Mostrar los números aleatorios
for(int i = 0; i < 8; i++)
{
    std::cout << "Random number " << i+1 << ": " << Array[i] << std::endl;
}

return 0;
}
```

- Ya que el tiempo de ejecución varía cada vez que se ejecuta el programa, los números generados tras la multiplicación serán diferentes cada vez.

```
C:\Users\JosU\Documents\CodeBlocks\RC4\bin\Debug\RC4.exe
Hardware info:
OEM ID: 9
Number of processors: 2
Page size: 4096
Processor type: 8664
Active processor mask: 3
Processor level: 6
Processor Architecture: 9
Processor Revisison: 5898

Time: 38002200 nanoseconds

Random number 1: 342019800
Random number 2: 76004400
Random number 3: 155657011200
Random number 4: 329251060800
Random number 5: 114006600
Random number 6: 228013200
Random number 7: 342019800
Random number 8: 224136975600

Process returned 0 (0x0)   execution time : 0.084 s
Press any key to continue.
```

```
C:\Users\JosU\Documents\CodeBlocks\RC4\bin\Debug\RC4.exe
Hardware info:
OEM ID: 9
Number of processors: 2
Page size: 4096
Processor type: 8664
Active processor mask: 3
Processor level: 6
Processor Architecture: 9
Processor Revisison: 5898

Time: 16000900 nanoseconds

Random number 1: 144008100
Random number 2: 32001800
Random number 3: 65539686400
Random number 4: 138631797600
Random number 5: 48002700
Random number 6: 96005400
Random number 7: 144008100
Random number 8: 94373308200

Process returned 0 (0x0)   execution time : 0.085 s
Press any key to continue.
```

```
C:\Users\JosU\Documents\CodeBlocks\RC4\bin\Debug\RC4.exe
Hardware info:
OEM ID: 9
Number of processors: 2
Page size: 4096
Processor type: 8664
Active processor mask: 3
Processor level: 6
Processor Architecture: 9
Processor Revisison: 5898

Time: 20001200 nanoseconds

Random number 1: 180010800
Random number 2: 40002400
Random number 3: 81924915200
Random number 4: 173290396800
Random number 5: 60003600
Random number 6: 120007200
Random number 7: 180010800
Random number 8: 117967077600

Process returned 0 (0x0)   execution time : 0.065 s
Press any key to continue.
```