SC

# Simulación

## Generadores

**Sergio H. Castro**

**2017**

SC

Guía de Trabajos prácticos utilizada en la asignaturas como Investigación Operativa, Métodos y Modelos Cuantitativos, Sistemas de Gestión de la Facultad Regional Córdoba Universidad Tecnológica Nacional y de la Facultad de Educación a Distancia del Instituto Universitario Aeronáutico que incluye temas de Métodos y Modelos para la Toma de Decisiones

# MÉTODOS Y MODELOS PARA SIMULACIÓN

## Contenido

## UNIDAD 2: GENERADORES DE NUMEROS ALEATORIOS

En cualquier experimento de simulación, así como en la mayoría de los experimentos de muestreo, existe la necesidad de contar con una fuente de números aleatorios. Los números aleatorios que estudiaremos en este capítulo se refieren a números o valores de variables que siguen la distribución uniforme. La finalidad del presente capítulo es analizar las propiedades convenientes de los números aleatorios utilizados en experimentos de simulación, a fin de presentar algunos métodos que se pueden utilizar para generar esos números aleatorios y, finalmente, presentar varias pruebas estadísticas que se pueden aplicar a un determinado conjunto de esos números aleatorios para determinar cuáles son las propiedades que poseen.

### 1. Generación de Números Aleatorios

Existen diferentes métodos utilizados para obtener números aleatorios, como ser:

**a. Métodos manuales**
**b. Tablas de números aleatorios.**
**c. Métodos de computadora analógica**.
**d. Métodos de computadora digital**

a. **Métodos manuales**. Se pueden obtener números aleatorios extrayendo (con reposición) bolillas numeradas consecutivamente desde un bolillero, arrojando dados, sacando con reposición naipes desde una baraja, leyendo las patentes de los autos, etc.

b. **Tablas de números aleatorios.** Varios libros de texto de estadística contienen grandes tablas de números aleatorios que se pueden utilizar conforme se haga necesario. Seguidamente se presenta una de ellas:

| | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|
| 51772 | 74640 | 42331 | 29044 | 46621 | 62898 | 93582 | 04186 | 19640 | 87056 |
| 24033 | 23491 | 83587 | 06568 | 21960 | 21387 | 76105 | 10863 | 97453 | 90581 |
| 45939 | 60173 | 52078 | 25424 | 11645 | 55870 | 56974 | 37428 | 93507 | 94271 |
| 30586 | 02133 | 75797 | 45406 | 31041 | 86707 | 12973 | 17169 | 88116 | 42187 |
| 03585 | 79353 | 81938 | 82322 | 96799 | 85659 | 36081 | 50884 | 14070 | 74950 |
| | | | | | | | | | |
| 64937 | 03355 | 95863 | 20790 | 65304 | 55189 | 00745 | 65253 | 11822 | 15804 |
| 15630 | 64759 | 51135 | 98527 | 62586 | 41889 | 25439 | 88036 | 24034 | 67283 |
| 09448 | 56301 | 57683 | 30277 | 94623 | 85418 | 68829 | 06652 | 41982 | 49159 |
| 21631 | 91157 | 77331 | 60710 | 52290 | 16835 | 48653 | 71590 | 16159 | 14676 |
| 91097 | 17480 | 29414 | 06829 | 87843 | 28195 | 27279 | 47152 | 35683 | 47280 |

c. **Métodos de computadora analógica**. Consiste en la generación de números aleatorios mediante el uso de computadoras analógicas diseñadas específicamente para ello. Esta técnica para la creación de variables aleatorias está actualmente en desuso.

d. **Métodos de computadora digital**. existen tres métodos para producir números aleatorios mediante un computador: Provisión externa, Generación interna a través de un proceso físico aleatorio y generación por medio de una regla de recurrencia o matemáticas. Son los generados mediante relaciones matemáticas (métodos aritméticos) que pueden ser manifestadas o representadas por medio de una serie de proposiciones en lenguaje de computadoras digitales, las cuales cuando son ejecutadas, causen que un número sea creado a partir de la distribución uniforme de probabilidad (denominados números pseudoaleatorios).

## 2. Generación de Números Pseudoaleatorios

Como se describió anteriormente estos métodos se realizan a través de relaciones que pueden expresarse de diferente manera la cual depende de la forma en que se obtiene.

Se llama números pseudoaleatorios a una sucesión determinística de números en el intervalo [0,1] que tiene las mismas propiedades estadísticas que una sucesión de números aleatorios. Una forma general de obtener números pseudoaleatorios es partir de una semilla de (p) números $u_{-p+1}, u_{-p+2}, \ldots, u_{-1}, u_0$ y aplicar una función (g) de modo que $u_i = g(u_{i-1}, \ldots, u_{i-p})$

Los números pseudoaleatorios son necesarios cuando se pone en práctica un modelo de simulación, para obtener observaciones aleatorias a partir de distribuciones de probabilidad.

Los números aleatorios generados en un inicio por una computadora casi siempre son números aleatorios enteros.

En sentido estricto, los números generados por una computadora no se deben llamar números aleatorios porque son predecibles y se pueden reproducir, dado el número aleatorio generador que se use. Por ello en ocasiones se les llama números pseudoaleatorios.

No obstante, el punto importante es que, en forma satisfactoria, hacen las veces los números aleatorios en la simulación si el método que se usa para generarlos es válido.

El procedimiento usado por una computadora para generar números aleatorios se llama generador de números aleatorios.

Un generador de números aleatorios es un algoritmo que produce secuencias de números que siguen una distribución de probabilidad específica y tienen la apariencia de aleatoriedad.

Un generador de números aleatorios consiste en una función que devuelve los valores de una secuencia de números reales, (x1, x2, . . . , xn), donde cada xi Є [0, 1]. El primer elemento de la secuencia se le denomina semilla

inicial de la serie, y a partir de ella debe ser posible generar el resto de la secuencia para que esta sea reproducible.

La semilla es el estado inicial del generador de números, es el $X_0$ que permite que la secuencia generada por la función sea distinta cada vez que se genere. Generalmente se utiliza el número de milisegundos desde Enero de 1970 (Unix epoch), o también se puede utilizar una semilla generada por ruido ambiental. Esto puede ser cualquier cosa desde el clima, el sonido, los electrones producidos por algún material radioactivo, etc.

La referencia a secuencias de números aleatorios significa que el algoritmo produce muchos números aleatorios en serie.

**La secuencia de números generados debe cumplir con las 2 hipótesis siguientes:**

1: Distribución Uniforme
2: Independencia (no correlacionados)

Además son importantes los siguientes aspectos:

1. Las subsecuencias también deben cumplir 1: y 2:
2. deben ser secuencias largas y sin huecos (densas)
3. algoritmos rápidos y que no ocupen mucha memoria.

**Los números aleatorios se pueden dividir en dos categorías principales:**

- Números aleatorios enteros. Es una observación aleatoria de una distribución uniforme discretizada en el intervalo n, n+1… Por lo general luego se transforma ese número a valores entre 0 ó 1 donde estos son valores convenientes para la mayoría de las aplicaciones.
- Números aleatorios uniformes. Es una observación aleatoria a partir de una distribución uniforme (continua) en un intervalo [a,b]

## 3. Propiedades a satisfacer por los números aleatorios y pseudoaleatorios:

**Propiedades mínimas que deberán satisfacer los números pseudoaleatorios:**

- Ajustarse a una distribución Uniforme. Podría ser $U(0;1)$, $U[0;1)$, $U(0;1]$, $U[0;1]$.
- Ser estadísticamente independientes (no debe deducirse un número conociendo otros ya generados).
- Ser reproducibles (la misma semilla debe dar la misma sucesión).
- Ciclo repetitivo muy largo.
- Facilidad de obtención.
- Ocupar poca memoria.

Cualquiera que sea el método para generar números aleatorios debe satisfacer las siguientes condiciones:

**Deben ser:**

1. **Uniformemente distribuidos**
2. **Estadísticamente independientes**
3. **Reproducibles**
4. **Sin repetición dentro de una longitud determinada de la sucesión**
5. **Generación a grandes velocidades**
6. **Requerir el mínimo de capacidad de almacenamiento**

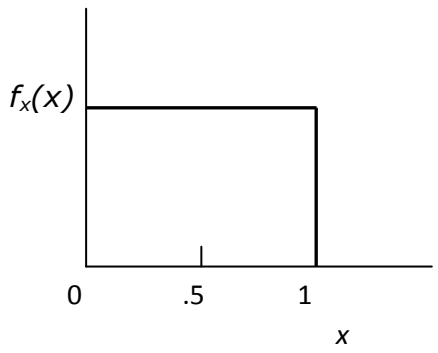
Para nuestros fines, debemos tener en consideración esas propiedades en función de un programa de computadora. Sencillamente, preguntémonos: **"¿cuáles son las propiedades convenientes de este programa de computadora para generar variables aleatorias uniformemente distribuidas?".**

Evidentemente, la primera respuesta que ofreceremos para esa pregunta es que la serie de números que produce ese generador debe seguir la distribución uniforme ideal tan de cerca como sea posible.

La pregunta a la que debemos responder es: **"¿qué propiedades debe poseer esa serie de variables aleatorias uniformemente distribuidas?".**

**Propiedades de los números aleatorios distribuidos uniformemente**

En primer lugar, examinaremos la distribución uniforme de probabilidad y definiremos las propiedades que posee. En la siguiente figura se muestra la función densidad para la distribución uniforme definida sobre el intervalo (0,1).




Para los fines de este análisis, a menos que se indique otra cosa, suponemos que nos ocupamos de la distribución uniforme continua definida sobre el intervalo (0,1).

La media de esta distribución debe ser ½. Además, para cualquier muestra dada, el segundo momento o la suma promedio de los cuadrados debe ser 1/3 y el tercer momento o la suma promedio de los cubos debe ser ¼.

Si tuviéramos que dividir el intervalo (0,1) entre *n* clases o subintervalos, el *i*ésimo subintervalo debería contener *T/n* observaciones, siendo T el número total de observaciones hechas. O sea, si hubiéramos hecho 1000 observaciones de las variables a partir de la distribución uniforme, poniéndolas en una distribución de frecuencias que contuviera, por ejemplo, 10 intervalos espaciados por igual, podríamos esperar tener una frecuencia de 1000/10 = 100 observaciones en cada intervalo. Esta propiedad debe ser válida, sea cual sea el tamaño de los intervalos. Es decir, si tuviéramos que reducir el intervalo en el caso discreto a un valor simple, podríamos esperar tener exactamente el mismo número de observaciones registradas para cada valor en el rango.

Esta propiedad se puede extender todavía más, diciendo que la probabilidad de observar un valor en un intervalo dado permanece constante y es independiente del valor obtenido anteriormente. Veamos nuevamente esta propiedad dada en función de la distribución discreta. Indica que, en el caso discreto, la probabilidad de que salga cualquier número dado es exactamente la misma para cada número diferente e independiente del número registrado anteriormente. O sea:

$$P(x_i = X) = \text{constante} \qquad \text{para todos los } i$$

Es preciso recordar que nos ocupamos de una serie de números que se generan sucesivamente en el tiempo. Teóricamente, debemos esperar eventos independientes y para cualquier intervalo dado, una probabilidad constante de que un valor dado caerá en ese intervalo. Puede resultar que nuestro generador particular no produzca de hecho esos números independientes y tenga una variación cíclica, por ejemplo se generen varios números por debajo de la media, y así sucesivamente. Las pruebas normales de frecuencia pueden indicar un buen ajuste; sin embargo, al tener en cuenta el orden en que aparecen los números, podremos ver que el generador no era verdaderamente aleatorio y que, por ende, no creamos eventos verdaderamente independientes.

Puesto que esas propiedades existen de manera ideal en cualquier nivel de significancia, deberemos poder repartir los números y comprobar la aleatoriedad de cada dígito o grupo de dígitos. Si se desean números aleatorios con tres posiciones decimales, será útil probar el generador hasta cinco posiciones decimales. De todos modos, las series de números que aparecen deben comprobarse en cada dígito si la aplicación a que se destina así lo requiere. En cualquier dígito dado del número aleatorio, la probabilidad de que aparezca los dígitos cero a nueve es de 0.1. Por ejemplo, la probabilidad de observar el número tres en el segundo lugar decimal de nuestro número aleatorio es 0.1 para cada número generado. Esta propiedad, como ya se indicó, debe comprobarse en todos los dígitos de cada número.

**Otras propiedades mínimas que deberán satisfacer los números pseudoaleatorios:**

Supongamos que el generador de números aleatorios produce números aceptables para nuestra aplicación. ¿Qué otras propiedades podemos esperar razonables que tenga este programa?. Las otras propiedades convenientes se dan a continuación:

a. **Debe ser rápido**; o sea, que debe generar un número en el menor tiempo posible. Este generador de números aleatorios es tan sólo una faceta de un modelo de simulación en computadora y, por ende, no debe consumir mucho tiempo.

b. **El programa mismo debe ser breve**. Esto quiere decir que no se requiera gran espacio de almacenamiento del mismo.

c. **Debe tener un período largo**. El período de un generador de números aleatorios es una medida de la cantidad de números que se generan, antes de que reaparezca la misma secuencia de números. El generador no se reciclará necesariamente mediante un regreso al primer número generado. Por ejemplo, el generador puede reciclarse y comenzar a reproducir la serie que se inicia en el décimo número generado.

d. Puesto que podemos desear duplicar el experimento varias veces, el generador **debe poder reproducir las mismas series de números** que se desee. Por otra parte, debe tener capacidad para producir, a voluntad, un conjunto o una serie claramente distintos.

e. **El generador debe ser de naturaleza no degenerativa.** La degeneración significa que el generador produce continuamente el mismo número. Si el generador tiene degeneración, el programa debe poder hacer correcciones y seguir adelante.

f. **Debe generar números estadísticamente independiente**,

Así pues, en general, podemos decir que un generador de números aleatorios debe ser un programa breve y rápido que produzca una larga secuencia de números aleatorios (que pasen las pruebas estándar) antes de comenzar a reciclarse y que tenga una naturaleza algorítmica. Como veremos, para cualquier experimento dado puede haber varios programas distintos que satisfagan esos criterios.

Al principio, se mencionó que un generador de números aleatorios era casi siempre un programa o subrutina de computadora. Debe resultar evidente que, si sólo se requieren cinco números aleatorios para un experimento dado de simulación, sería mucho más económico leerlos de archivo, después de tomar los valores de un conjunto de tablas. Además, si un generador dado de números aleatorios llega a generar perfectamente 30 números aleatorios antes de reciclarse (o sea, si tiene un período de 30 números) y sólo necesitamos 20 para nuestro experimento, ese generador será apropiado para nuestros fines.

Antes indicamos que el generador debe ser de naturaleza algorítmica. Se trata de un requisito muy práctico para el programa generador. El que sea algorítmico quiere decir que se utiliza el resultado del cálculo anterior para determinar el siguiente. De hecho, se trata de una cualidad útil, porque le

permite al generador funcionar independientemente de todas las demás partes del programa de simulación. En general, lo único que se debe hacer es proporcionar un valor inicial o de partida al generador, que sigue adelante, generando una secuencia de valores a partir de ese inicial.

Para ser precisos, debemos observar que los métodos presentados aquí para la creación de generadores de números aleatorios son determinísticos, puesto que todos ellos incluyen una técnica repetitiva o una relación de congruencia, expresada por medio de una fórmula. Cualquier serie de números creados de este modo no podrá ser "verdaderamente aleatoria". En teoría, sólo algunos fenómenos físicos pueden ser procesos aleatorios verdaderos. En vista de esto, muchos autores hablan de "generadores de números pseudoaleatorios" en lugar de "generadores de números aleatorios".

## 4. Métodos de Generación de Números Pseudoaleatorios

Definición Formal: Un generador de números (pseudo)aleatorios es una estructura G = (X; $x_0$; T; U; g), donde X es un conjunto finito de estados, $x_0$ Є X es el estado inicial (semilla), la aplicación T : X → X es la función de transición, U es el conjunto finito de posibles observaciones, y G : X → U es la función de salida.

El funcionamiento de un generador de números pseudo-aleatorios sería el siguiente:

- Se elige una semilla inicial cualquiera $x_0$, y se genera una sucesión de valores $x_n$ mediante una relación de recurrencia $x_n = T(x_{n-1})$.
- Cada uno de estos valores proporciona un número pseudo-aleatorio $u_n$ definido a través de alguna relación $u_n = g(x_n)$
- Claramente, la sucesión de estados es periódica, puesto que X es finito.
- En algún momento, ocurrirá que $x_j = x_i$ para algún j > i, y a partir de ese instante, $x_{j+k} = x_{i+k}$, y por lo tanto, $u_{j+k} = u_{i+k}$, para todo k>= 0.

El periodo es el menor entero ρ>0 tal que para algún entero τ >=0, se verifica que $x_{\rho+k} = x_k$, para todo k >= τ.

Claramente, el periodo de un generador no puede exceder el cardinal del espacio de estados.

Una buena propiedad para un generador es que su periodo este cercano a $|X|$.

Podemos observar diferentes modelos generales para la obtención de números pseudoaleatorios

### 4.1. Métodos Aritméticos para generar Números Aleatorios

#### Técnica de "cuadrado medio" (Midsquare Method)

El primer método para la generación de números aleatorios se conoce como técnica de **"cuadrado medio" (midsquare).** Lo propuso John Von

Neumann en una conferencia de 1949 y Nicolás Metropolis hacia 1951 reporto secuencias de hasta 750 mil dígitos antes terminar la secuencia.

En el método de los cuadrados medios, el proceso para la generación consiste en tomar un número al azar de 2n dígitos, que al elevarlo al cuadrado resulta un numero de 4n dígitos y cada número sucesivo se genera tomando los 2*n* dígitos centrales del cuadrado del número anterior de 2*n* dígitos. Sea x1 el numero resultante de seleccionar las 2n cifras centrales del cuadrado de $x_0$; el primer número aleatorio $u_1$ se obtiene poniendo un punto decimal delante las 2n cifras de $x_1$. A continuación $x_2$ y $u_2$ se generan a partir de x1 del mismo modo. As$_i$ sucesivamente.

$$x_{m+1} = (x_m)^2 \quad 2\text{n dígitos medios}$$

Para generar una secuencia de números pseudoaleatorios de 4 dígitos, se crea un valor de partida de 4 dígitos y se lo eleva al cuadrado, produciendo un número de 8 dígitos. Si el resultado es menor de 8 dígitos, se agregan ceros a la izquierda para compensar. Los 4 dígitos del medio del resultado serían el siguiente número de la secuencia, y obtiene como resultado. Este proceso se repite para generar más números.

Por ejemplo, supóngase que deseamos generar números aleatorios de 4 dígitos (2x2 dígitos). Nuestro primer valor es 4122. Esto nos da la secuencia siguiente de números:

4122➔9908➔1684➔8358➔8561➔2907➔4506➔3040.

Los problemas que se presentan en este método se deben a que:

- tiende a degenerar con rapidez a cero. Dependiendo del valor inicial, el método puede degenerar al cabo de 20 términos.
- en este método se producirá eventualmente la repetición de algunas series y la secuencia de números aleatorios se reciclará.
- debido a las operaciones que se deben realizar este método no es muy rápido (por ejemplo pruebe con el número 3708).

Analicemos otros ejemplos:

| **Análisis del Proceso: (2n=2)** | | | | |
|---|---|---|---|---|
| **$X_m$** | **$(X_m)^2$** | **Valores Medios** | **$X_{m+1}$** | **Aleatorio Decimal** |
| 87 | 7569 | 56 | **56** | **0,56000** |
| 56 | 3136 | 13 | **13** | **0,13000** |
| 13 | 169 | 16 | **16** | **0,16000** |
| 16 | 256 | 25 | **25** | **0,25000** |
| 25 | 625 | 62 | **62** | **0,62000** |
| 62 | 3844 | 84 | **84** | **0,84000** |
| 84 | 7056 | 5 | **5** | **0,05000** |
| 5 | 25 | 2 | **2** | **0,02000** |
| 2 | 4 | 0 | **0** | **0,00000** |

| | |
|---|---|
| Semilla | **87** |
| Cifras | **2** |
| Cifras Centrales Únicas | **Si** |

**Longitud del Generador: 10 números**

Si no se utilizaran 2n dígitos surge el siguiente problema con su respectiva solución:

| Análisis del Proceso | | | | |
|---|---|---|---|---|
| **X0** | **X*X** | **Valores Medios** | **$X_{m+1}$** | **Aleatorio Decimal** |
| 412 | 169744 | 974 | **974** | **0,97400** |
| 974 | 948676 | 867 | **867** | **0,86700** |
| 867 | 751689 | 168 | **168** | **0,16800** |
| 168 | 28224 | 822 | **822** | **0,82200** |

| | |
|---|---|
| Semilla | **412** |
| Cifras | **3** |
| Cifras Centrales Únicas | **No** |
| ¿Usa Centrales de Derecha? | **Si** |
| **Longitud del Generador: 15 números** | |

| Análisis del Proceso | | | | |
|---|---|---|---|---|
| **X0** | **X*X** | **Valores Medios** | **$X_{m+1}$** | **Aleatorio Decimal** |
| 412 | 169744 | 697 | **697** | **0,69700** |
| 697 | 485809 | 858 | **858** | **0,85800** |
| 858 | 736164 | 361 | **361** | **0,36100** |
| 361 | 130321 | 303 | **303** | **0,30300** |

| | |
|---|---|
| Semilla | **412** |
| Cifras | **3** |
| Cifras Centrales Únicas | **No** |
| ¿Usa Centrales de Derecha? | **No** |
| ¿Usa Centrales de Izquierda? | **Si** |
| **Longitud del Generador: 41 números** | |

## Técnica del "producto medio"

Otro método de generación de números aleatorios se denomina técnica del **producto medio** y es muy similar a la técnica de cuadrado medio en que el número aleatorio resultante se toma de los 2*n* dígitos centrales del resultado de una multiplicación previa. En notación matemática:

$$x_{m+1} = x_m \times x_{m-1} \quad 2\,\text{n}\ \text{dígitos}\ \ \text{medios}$$

La técnica para el producto medio implica la elección de dos números aleatorios $x_1$ y $x_2$, cada uno de ellos con 2n dígitos. Luego, se multiplica $x_1$ por $x_2$ para obtener *U*. Se hace $x_3$ igual a los *2n* dígitos centrales de *U*. A continuación $x_4$ es igual a $x_3$ multiplicado por $x_2$, y así sucesivamente.

| Análisis del Proceso (2n=4) | | | | | |
|---|---|---|---|---|---|
| **$X_{m-1}$** | **$X_m$** | **$X_{m-1}$*$X_m$** | **Valores Medios** | **$X_{m+1}$** | **Aleatorio Decimal** |
| 1374 | 2459 | 3378666 | 3786 | **3786** | **0,37860** |
| 2459 | 3786 | 9309774 | 3097 | **3097** | **0,30970** |
| 3786 | 3097 | 11725242 | 7252 | **7252** | **0,72520** |
| 3097 | 7252 | 22459444 | 4594 | **4594** | **0,45940** |
| 7252 | 4594 | 33315688 | 3156 | **3156** | **0,31560** |
| 1374 | 2459 | 3378666 | 3786 | **3786** | **0,37860** |

Semilla1 **1374**

Semilla2 **2459**

**Longitud del Generador: 41 números**

## Técnica del "producto medio" con multiplicador constante

Es una modificación del método del producto medio y consiste en utilizar un multiplicador constante, en lugar de números aleatorios; o sea:

$$x_{n+1} = (K \cdot x_n) \qquad n\ digitos\ medios$$

Recordemos que n es la suma de dígitos del número utilizado como semilla y de la constante utilizada.

Este método es similar al de cuadrado medio. No obstante, los dos tienen períodos más largos y los números que producen parecen estar distribuidos más uniformemente. Sin embargo, como sucede en el caso de la técnica de cuadrado medio esta técnica parece degenerar finalmente hacia algún valor constante. Tanto el método de cuadrados medios como el de producto medio tienen un período relativamente breve que se ve afectado considerablemente por los valores escogidos inicialmente.

| **Anallisis del Proceso** | | | | | | |
|---|---|---|---|---|---|---|
| **$X_m$** | **Cte K** | **$K*X_{m-1}$** | **Valores Medios** | **$X_{m+1}$** | **Aleatorio Decimal** | Semilla1 **3978** |
| 3978 | 7203 | 28653534 | 6535 | **6535** | **0,65350** | Semilla2 **7203** |
| 6535 | 7203 | 47071605 | 716 | **716** | **0,07160** | **Longitud del Generador: 988 números** |
| 716 | 7203 | 5157348 | 1573 | **1573** | **0,15730** | |
| 1573 | 7203 | 11330319 | 3303 | **3303** | **0,33030** | |
| 3303 | 7203 | 23791509 | 7915 | **7915** | **0,79150** | |

## Método de Lehmer

También podemos destacar el **método de Lehmer**; en este se parte de un numero al azar de n cifras, se le multiplica por un número al azar de K cifras, dando lugar a un número de n+K cifras de que se separan las K cifras de la izquierda, obteniéndose un número de n cifras del cual se resta el de K cifras que se había separado.

Ej.: $X_0$= 4122 K=76 4122x76 = 31|3272 3272-31 = 3241.

$X_1$= 3241 K=76 3241x76 = 24|6316 6316-24 = 6292.

$X_2$= 6292 K=76 6292x76 = 47|8192 8192-47 = 8145.

$X_3$= 8145 K=76 8145x76 = 61|9020 9020-61 = 8959.

$X_4$= 8959 K=76 8959x76 = 68|0884 0884-68 = 0816.

$X_5$= 0816 .....

## 4.2. Métodos de Congruencia o congruenciales

Los generadores congruenciales fueron descubiertos por Lehmer en 1951 cuando observo que los residuos de las potencias sucesivas de un número sugieren un comportamiento aleatorio.

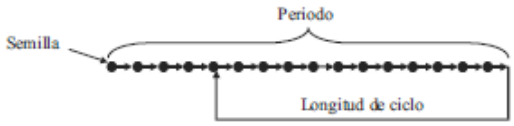
La primera propuesta de Lehmer consistió en la siguiente secuencia:

$$x_{n+1} = a^n \, mod \, m$$

Es decir, el número n-ésimo de la secuencia se obtiene dividiendo la potencia n-esima de un entero a por un entero m y tomando el resto. La expresión anterior es equivalente a la siguiente:

$$x_{n+1} = (a \cdot x_n) mod \, m$$

De este modo se puede obtener cada elemento de la secuencia a partir del elemento anterior, y en primer lugar de un valor inicial $x_0$ denominado semilla.




Muchos de los generadores propuestos actualmente son una generalización del propuesto por Lehmer, donde se incluye un sesgo b en la expresión anterior:

$$x_{n+1} = (c + a \cdot x_n) mod \, m$$

Con semilla $x_0$:

$$a \in \mathbb{Z}, \; 1 \leq a \leq m, \; x_0 \in \mathbb{Z}, \; 0 \leq x_0 \leq m-1, \; c \in \mathbb{Z}, \; 0 \leq c \leq m, \; m \in \mathbb{Z}.$$

Entonces se toma $r_i = \frac{x_i}{m} \in [0,1)$

La secuencia así obtenida consiste en una serie de números enteros comprendidos entre 0 y n − 1.

Las parámetros del generador a, b y m deben de ser números enteros positivos.

Por otro lado, cuando c = 0 diremos que el generador es multiplicativo.

### Conversión del número Generado a pseudoaleatorio entre 0 y 1

Para obtener una secuencia de números entre 0 y 1 a partir de un Generador congruencial lineal bastara con

- dividir cada $x_n$ por el modulo m de la serie: $r_n = x_n/m$ donde $r_n \in U[0;1)$
- dividir cada $x_n$ por el modulo (m-1) de la serie: $r_n = x_n/(m-1)$ donde $r_n \in U[0;1]$
- dividir cada $x_n$ por el modulo m de la serie: $r_n = (x_n+½)/m$ donde $r_n \in U(0;1)$

A pesar de su simplicidad, la selección adecuada de las constantes (a, c, m) permitirá obtener secuencias largas y aleatorias de un modo muy eficiente.

Los métodos mencionados anteriormente, que se solían utilizar ampliamente, han cedido su lugar a los **métodos de congruencia** basados en la relación:

$$x_{n+1} \equiv (a \cdot x_n + c) mod\ m \qquad 0 \leq x_n \leq m$$

Esta relación implica que la suma $ax_i+C$ se debe dividir por $m$ y $r_{i+1}$ es igual al residuo. La relación indica *"$x_{i+1}$ es congruente con $ax_i+C$ módulo $m$"*. Como ilustración, sea $m=25$, $a=6$ y $C=1$; asimismo, sea $x_0=1$.

$$x_1 \equiv 6 \times 1 + 1 (mód\ 25) \qquad x_1 \equiv 7$$

$$x_2 \equiv 6 \times 7 + 1 (mód\ 25) \qquad x_2 \equiv 18$$

$$x_3 \equiv 6 \times 18 + 1 (mód\ 25) \qquad x_3 \equiv 9$$

Se cuenta con varios generadores de números aleatorios, de los cuales los más populares son los métodos congruenciales (aditivo, multiplicativo y mixto).

## Método multiplicativo de congruencia

El método lo propuso inicialmente Lehmer en 1949. Con $C=0$, se denomina **método multiplicativo de congruencia** o método congruencial multiplicativo corresponde al caso especial del método congruencial mixto en el que c =0

$$x_{n+1} \equiv (a \cdot x_n) mod\ m$$

En este caso, los números $r_i$ pueden ser generados por la ecuación

$$r_n = \frac{x_n}{(m-1)} \qquad 0 \leq x_n < m \quad \forall n$$

De acuerdo con Banks, Carson, Nelson y Nicol, las condiciones que deben cumplir los parámetros para que el algoritmo congruencial multiplicativo alcance su máximo período son:

m debe ser múltiplo de $2^g$ , donde g debe ser entero, a = 3 + 8k , donde k=0, 1, 2, 3,... ; $X_0$ debe ser un número impar.

Bajo estas condiciones se logra un período de vida máximo: N= k/4= 2g−2.

## Método congruencial mixto

El método de congruencia que se muestra con C≠0 se denomina **método de congruencia mixta** o congruencial mixto y genera una sucesión de números aleatorios enteros en un intervalo de 0 a m-1. Éste método siempre calcula el siguiente número a partir del último que obtuvo, dado un número aleatorio inicial Xo, llamado semilla. En particular, calcula el (n+1)ésimo número aleatorio Xn+1 a partir del n-ésimo número aleatorio Xn con la relación de recurrencia.

$$x_{n+1} \equiv (a \cdot x_n + c) mod\ m \qquad 0 \leq x_n \leq m$$

donde

- X es la secuencia de números pseudoaleatorios, y
- a = es la constante multiplicativa.
- c = es la constante aditiva.
- m = es la magnitud del módulo.
- X0 = es la semilla.

Aquí mod representa a la operación aritmética módulo entre los enteros a y b tal que el resultado de (a mod b) es el residuo entero de la división a entre b. Por ejemplo 16 mod 3 es igual a 1, ya que 16 dividido 3 tiene 1 como residuo entero de la división.

Donde a, c y m son enteros positivos (a < m, c < m, $x_0$ < m). Ésta notación matemática significa que $X_{n+1}$ son 0, 1, …, M-1, de manera que m representa el número deseado de valores diferentes que se puede generar como números aleatorios.

A manera de ilustración, suponga que m=99, a=71, c=67 y Xo=99. En la siguiente tabla se calculó los primeros de la sucesión de números aleatorios que se tuvo (esta sucesión tiene una longitud de 99 números hasta que comiencen a repetirse los números en el mismo orden).

| | | | |
|---|---|---|---|
| Semilla | **0** | Const. A | **67** |
| Const. C | **71** | Const. M | **99** |

| **Análisis del Proceso** | | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| **C** | **A** | **Xi** | **A*Xi** | **C + A*Xi** | **M** | **(C+ A*Xi) / M** | **Parte entera** | En Fracciones | | **Parte entera * M** | **Resto o Residuo** | **Ri+1** |
| 71 | 67 | 0 | 0 | 71 | 99 | 0,717 | 0 | 0 | 71/99 | 0 | 71 | 71 |
| 71 | 67 | 71 | 4757 | 4828 | 99 | 48,767 | 48 | 48 | 76/99 | 4752 | 76 | 76 |
| 71 | 67 | 76 | 5092 | 5163 | 99 | 52,151 | 52 | 52 | 5/33 | 5148 | 15 | 15 |
| 71 | 67 | 15 | 1005 | 1076 | 99 | 10,868 | 10 | 10 | 86/99 | 990 | 86 | 86 |
| 71 | 67 | 86 | 5762 | 5833 | 99 | 58,919 | 58 | 58 | 91/99 | 5742 | 91 | 91 |
| 71 | 67 | 91 | 6097 | 6168 | 99 | 62,303 | 62 | 62 | 10/33 | 6138 | 30 | 30 |
| 71 | 67 | 30 | 2010 | 2081 | 99 | 21,020 | 21 | 21 | 2/99 | 2079 | 2 | 2 |
| 71 | 67 | 2 | 134 | 205 | 99 | 2,0707 | 2 | 2 | 7/99 | 198 | 7 | 7 |
| 71 | 67 | 7 | 469 | 540 | 99 | 5,4545 | 5 | 5 | 5/11 | 495 | 45 | 45 |

**Longitud del Generador: 99 números (repite el cero luego de generar 99 numeros. Hay que recordar que genera valores entre 00 y 98 porque usa un módulo de 99)**

Nótese que si arma una tabla con los 99 números mencionados y la compara con los valores obtenidos si hubiéramos usado le numero 71 como semilla, obtendríamos la misma secuencia, solo que iniciada en otro valor y también seria de una longitud de 99.

En otro ejemplo pero buscando números de una cifra, podríamos utilizar los siguientes valores: m=9, a=4, c=5 y Xo=1. En la siguiente tabla se calculó los primeros de la sucesión de números aleatorios que se tuvo (esta sucesión tiene una longitud de 100 números hasta que comiencen a repetirse los números en el mismo orden).

| | | | |
|---|---|---|---|
| Semilla | 7 | Const. C | 5 |
| | | Const. A | 4 |
| Const. M | 9 | | |

**Análisis del Proceso**

| xi | A*xi | C + A xi | (C + A xi) / M | En Fracciones | Parte entera | Parte entera * M | Resto o Residuo | $x_{i+1}$ |
|---|---|---|---|---|---|---|---|---|
| 7 | 28 | 33 | 3,666 | 3 2/3 | 3 | 27 | 6 | 6 |
| 6 | 24 | 29 | 3,222 | 3 2/9 | 3 | 27 | 2 | 2 |
| 2 | 8 | 13 | 1,444 | 1 4/9 | 1 | 9 | 4 | 4 |
| 4 | 16 | 21 | 2,333 | 2 1/3 | 2 | 18 | 3 | 3 |
| 3 | 12 | 17 | 1,888 | 1 8/9 | 1 | 9 | 8 | 8 |
| 8 | 32 | 37 | 4,111 | 4 1/9 | 4 | 36 | 1 | 1 |
| 1 | 4 | 9 | 1 | 1 | 1 | 9 | 0 | 0 |
| 0 | 0 | 5 | 0,555 | 5/9 | 0 | 0 | 5 | 5 |
| 5 | 20 | 25 | 2,777 | 2 7/9 | 2 | 18 | 7 | 7 |

**Longitud del Generador: 9 números**

La cantidad de números consecutivos en una sucesión antes de que se repita se conoce como longitud de ciclo. En consecuencia, la longitud de ciclo en el ejemplo es 9 (genera valores entre 0 y 8 porque usa un módulo de 9). La longitud de ciclo máxima es m, de manera que sólo los valores de a y c considerados son los que conducen a una longitud de ciclo máxima.

El método congruencial mixto proporciona una gran flexibilidad para elegir un generador de números aleatorios en particular (una combinación específica de a, c y m). Sin embargo, se requiere tener mucho cuidado al seleccionar el generador de números aleatorios porque la mayoría de las combinaciones de valores a, c y m conducen a propiedades indeseables (por ejemplo, una longitud de ciclo menor a m).

Cuando se quiere construir un generador de números aleatorios para simular los valores de una variable aleatoria, se deben elegir los parámetros de tal manera que se garantice un periodo largo para que se puedan hacer todos los ensayos de simulación, por lo tanto se deben tener en cuenta las siguientes condiciones:

- a debe ser un número impar, no divisible ni por 3 ni por 5.
- c usualmente puede ser cualquier constante, sin embargo, para asegurar buenos resultados, se debe seleccionar a de tal forma que, a mod 8 = 5 para una computadora binaria, o a mod 200 = 21 para computadora decimal.

- m debe ser el número entero más grande que la computadora acepte.

De acuerdo con Hull y Dobell, los mejores resultados para un generador congruencial mixto en una computadora binaria son:

- c = 8*a±3
- a = cualquier entero
- $X_0$ = Cualquier entero impar.
- M = 2b donde b >2 y que m sea aceptado por la computadora.

**Longitud del Generador**

Si $X_i=X_{i+k}$ por vez primera, entonces k se denomina período del generador. Si c=0 el método se dice multiplicativo. A continuación, enunciamos dos resultados relacionados con congruencias cuya demostración se omite. Condición necesaria para que un generador multiplicativo tenga un período de longitud M-1 es que M sea primo. Si M es primo, el período divide a M-1.

En este caso, el período es M-1 si, y sólo si, $a^{\frac{M-1}{p}} \not\equiv 1(M) \text{ para ningún factor primo } p \text{ de } M-1.$

Sea a≠0. Un generador congruencial tiene período M si, y sólo si, se cumplen las tres condiciones siguientes:

1. $mcd(c,M)=1$.
2. $a \equiv 1(p)\ \forall p$ factor primo de M.
3. $a \equiv 1(4)$ si M es múltiplo de 4.

A continuación se presenta el generador implementado en los IBM380 en 1970. Este método resultó tener un gran problema: hay quince planos paralelos en el cubo $[0,1]^3$ que contienen todos los puntos que se obtienen tomando tres valores $X_i$, $X_{i+1}$, $X_{i+2}$ cualesquiera.

**Ejemplo:** c=0, $a=2^{16}+3$, $M=2^{31}$ Los $c_i$ que se utilizan a continuación representan enteros. Su valor exacto se omite pues no tiene interés.

$X_{i+1}=(2^{16}+3)\ X_i + 2^{31}\ c_0$

$$X_{i+2} = (2^{16}+3)X_{i+1} + 2^{31}c_1 = (2^{16}+3)^2 X_i + 2^{31}c_3 = (6\cdot 2^{16}+9)X_i + 2^{31}c_4 =$$

$=(6\ \ (2^{16}+3)-9)\ X_i + \ \ 2^{31}\ c_4 = \ \ 6\ \ (2^{16}+3)X_i - \ \ 9\ X_i + \ \ 2^{31}\ c_4 = \ \ 6\ X_{i+1} - 9\ X_i + 2^{31}\ c_5$

Así $u_{i+2} - 6u_{i+1} + 9u_i = c_5 \in \mathbb{Z}.$

Como $u_i$, $u_{i+1}$, $u_{i+2} \in [0,1)$ y $c_5 \in \mathbb{Z}$, enonces $c_5 \in \{-5,-4,\ldots,8,9\}$. Es decir, todas las ternas$(u_i,u_{i+1},u_{i+2})$ están concentradas en quince planos.

Un generador recomendado es $X_{i+1} = (69069X_i + 1) \equiv 2^{32}.$

## Método congruencial aditivo

El método congruencial aditivo también es parecido, pero establece a =1 y sustituye a c por algún número aleatorio anterior a Xn en la sucesión , por ejemplo, Xn-1 (así requiere más de una semilla para iniciar el cálculo de la sucesión).

$$x_{n+1} \equiv (x_n +\cdot x_{n-1}) mod\ m \qquad 0 \leq x_n < m$$

se conoce como **método congruente aditivo**. De hecho, se trata de una secuencia de Fibonacci cuando $r_0=0$ y $r_1=1$.

| Análisis del Proceso | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|
| $x_{n-1}$ | $x_n$ | $x_{n-1}+x_n$ | M | $\frac{(x_{n-1}+x_n)}{M}$ | En Fracciones | Parte entera | Parte entera * M | Resto o Residuo | $x_{n+1}$ |
| Semilla1 | Semilla2 | | | | | | | | |
| 244 | 706 | 950 | 901 | 1,054384 | 1 49/901 | 1 | 901 | 49 | 49 |
| 706 | 49 | 755 | 901 | 0,837958 | 755/901 | 0 | 0 | 755 | 755 |
| 49 | 755 | 804 | 901 | 0,892342 | 804/901 | 0 | 0 | 804 | 804 |
| 755 | 804 | 1559 | 901 | 1,7303 | 1 658/901 | 1 | 901 | 658 | 658 |
| 804 | 658 | 1462 | 901 | 1,622642 | 1 33/53 | 1 | 901 | 561 | 561 |
| 658 | 561 | 1219 | 901 | 1,352941 | 1 6/17 | 1 | 901 | 318 | 318 |
| 561 | 318 | 879 | 901 | 0,975583 | 879/901 | 0 | 0 | 879 | 879 |
| 318 | 879 | 1197 | 901 | 1,328524 | 1 296/901 | 1 | 901 | 296 | 296 |

**Longitud del Generador: 100números**

En este caso, los números $r_i$ pueden ser generados por la ecuación

$$r_n = \frac{x_n}{(m-1)} \qquad 0 \leq x_n < m \quad \forall n$$

## Longitud del Ciclo de un Generador

La cantidad de números consecutivos en una sucesión antes de que se repita se conoce como longitud de ciclo.

Por ejemplo, si identificamos un Generador lineal congruencial general

$$X_i = (a_1 X_{i-1} + \mathrm{K} + a_p X_{i-p}) \bmod m$$
$$p > 1,\ a_p \neq 0$$

No hay repetición hasta que $(X_0, \mathrm{K}, X_{p-1}) = (X_k, \mathrm{K}, X_{k+p-1})$

Potencialmente, período máximo $m^p$-1.

$p$=2, $a_1$=$a_2$=1: de Fibonacci (muy malos)

## Consideraciones sobre los Métodos congruenciales

Se ha demostrado que el método congruente multiplicativo, cuando:

$$x_{n+1} \equiv (a \cdot x_n) mod\ m$$

se comporta muy bien desde el punto de vista estadístico (Gorenstein, 1967).

Por lo común, en una computadora binaria se considera que *m* es alguna potencia de *2*. En este mismo tipo de computadoras, el periodo máximo es *m/4*, donde $m=2^b$ *(b>2)* y se logra con $r_0$ impar y *a = 8t±3*, donde *t=1,2,3*. En este método suele ser conveniente hacer que *b* tome el valor del número de bits de una palabra binaria de la computadora que se utilice.

Un avance respecto a lo anterior fue realizado por MacLaren y Marsaglia (1965) que propusieron una combinación de dos generadores congruentes para producir secuencias aleatorias. En este método se utiliza un generador para mezclar la secuencia producida por el otro.

Los métodos analizados corresponden a los primeros desarrollos. En la actualidad existen gran cantidad de ellos y día a día aparecen nuevos métodos cada vez más complejos y sofisticados.

## Algoritmos Congruenciales Cuadrático (no lineal)

Este algoritmo tiene la ecuación recursiva:

$$x_{n+1} \equiv (a \cdot x_n^2 + b \cdot x_n + c) mod\ m \qquad 0 \leq x_n < m \ \ con\ n = 0, 1, 2, \dots, k$$

En este caso, los números $r_i$ pueden ser generados por la ecuación

$$r_n = \frac{x_n}{(m-1)} \qquad 0 \leq x_n < m \quad \forall n$$

De acuerdo con L'Ecuyer, las condiciones que deben cumplir los parámetros m , a , b y c para alcanzar un período máximo de N = m son: m debe ser múltiplo de $2^g$ , donde g debe ser entero, a debe ser un número par, m debe ser un número impar, y (b −1)mod 4 = 1. De esta manera se logra un período de vida máximo N = m.

## Algoritmos Congruencial de Blum, Blum, y Shub (no lineal)

Dentro de los algoritmos congruenciales no lineales se tiene el algoritmo congruencial cuadrático y el de Blum, Blum, y Shub.

## Combinación de algoritmos congruenciales

Los algoritmos congruenciales se pueden combinar para aumentar el periodo del ciclo de generación. Estas combinaciones se basan en los siguientes resultados:

- Si U1, … Uk son variables aleatorias U(0;1), entonces la parte fraccional de U1 + … Uk también sigue una distribución U(0;1)
  U1 + U2 + … + Uk - [U1 + U2 + … Uk] ~ U(0; 1)
- Si u1; u2;…; uk están generados por algoritmos congruenciales con ciclos de periodo c1; c2;…; ck, respectivamente, entonces la parte fraccional de $u_1+u_2+\dots+u_k$ tiene un ciclo de periodo m.c.m. $\{c_1; c_2;\dots; c_k\}$.

El algoritmo combinado de Wichmann y Hill (1982,1984) tiene un periodo de orden $10^{12}$. El generador es:

$$x_i \equiv (171 \cdot x_{i-1}) mod\ 30269$$

$$y_i \equiv (172 \cdot y_{i-1}) mod\ 30307$$

$$z_i \equiv (170 \cdot z_{i-1}) mod\ 30323$$

y tomar

$$u_i \equiv \left(\frac{x_i}{30269} + \frac{y_i}{30307} + \frac{z_i}{30323}\right) - \left[\frac{x_i}{30269} + \frac{y_i}{30307} + \frac{z_i}{30323}\right]$$

## 4.3. Otros generadores

Aunque los generadores congruenciales son los más utilizados en la práctica, se han desarrollado otros tipos de generadores con la intención de obtener periodos más largos y mejores propiedades estadísticas. A menudo, sin embargo, un generador congruencial con parámetros elegidos adecuadamente puede funcionar tan bien que otras alternativas más complicadas. Los generadores congruenciales pueden generalizarse a recursiones lineales de orden mayor, considerando la siguiente relación

$$x_{n+1} \equiv (a_1 \cdot x_{n-1} + \cdots + a_k \cdot x_{n-k}) mod\ m$$

donde el orden k y el modulo m son enteros positivos, y los coeficientes aj son enteros variando entre -(m-1) y (m-1). En la n-ésima iteración, el estado es el vector $(x_n; x_{n-1}; \ldots; x_{n+k-1}) \in Z_m^k$ . La función de salida se puede definir simplemente como $u_n = x_n/m$.

El estudio de este generador se asocia al estudio del polinomio característico

$$P(z) = z^k - a_1 z^{k-1} - \ldots - a_k$$

sobre el cuerpo finito $Z_m$. Cuando m es primo y el polinomio es primitivo sobre $Z_m$, el periodo del generador es $m^k-1$ (periodo máximo posible en esta clase de generadores).

### Generadores de desplazamiento de bits

En estos generadores cada nuevo número entero aleatorio $x_i$, se obtiene manipulando los bits del número anterior, $x_{i-1}$. En lenguaje C, esto se puede hacer fácilmente utilizando operadores sobre bits, >>, <<, ^, |, &.

### Generadores de Fibonacci

Las grandes ventajas de estos generadores es que son generadores muy rápidos que tienen un periodo muy largo. La fomentación teórica en la que se basan es diferente a la de los GCL. Los generadores de Fibonacci se basan en una recurrencia del tipo

$$N_i = (N_{i-r} \circ N_{i-s}) \bmod m$$

Donde r<s son enteros dados y ° denota alguna de las operaciones +, -, x, ^. Este tipo de generador precisa iniciar (con otro generador) y mantener una lista de los últimos s números generados.

## Generador frac

Sean $\{u_i\}_i$ $\{v_i\}_i$,
dos sucesiones de números pseudoaleatorios, generados congruencialmente, con respectivos períodos $c_1$ y $c_2$.

La sucesión $\{frac(u_i + v_i)\}_i$ tiene período *mcm*($c_1$,$c_2$).

Sean $U_1$, $U_2$ variables aleatorias independientes con distribución U(0,1).

La variable $Z=frac(U_1+U_2)$ sigue también una distribución U(0,1).

**Demostración:**

Como $U_1$, $U_2$ son independientes, entonces la variable bidimensional ($U_1$,$U_2$) sigue una distribución uniforme en el cuadrado $[0,1] \times [0,1]$.

$$f_{(U_1,U_2)}(u_1, u_2) = 1 \text{ en } [0,1] \times [0,1].$$

Sean $\left.\begin{array}{rcl} X & = & U_1 + U_2 \\ Y & = & U_2 \end{array}\right\}$

$$\left.\begin{array}{rcl} U_1 & = & X - Y \\ U_2 & = & Y \end{array}\right\} \Rightarrow \left|\frac{\partial(u_1, u_2)}{\partial(x, y)}\right| = \left|\begin{array}{cc} 1 & -1 \\ 0 & 1 \end{array}\right| = 1$$

$$f(x, y) = 1 \text{ en } \; R, \text{ con } \; R = \{(x, y) \; / \; 0 \leq y \leq 1, \; y \leq x \leq 1 + y\}.$$

La distribución marginal es

$$f_X(x) = \left\{\begin{array}{ccc} x & si & 0 \leq x \leq 1 \\ 2 - x & si & 1 \leq x \leq 2 \end{array}\right.$$

Por otra parte,

$$Z = frac(X) = \left\{\begin{array}{ccc} X & si & 0 \leq X \leq 1 \\ 2 - X & si & 1 \leq X \leq 2 \end{array}\right.$$

Así, para $Z=frac(X)=\begin{cases} X & \text{si } 0\le X\le 1 \\ X-1 & \text{si } 1\le X\le 2 \end{cases}$, se tiene

$$F_Z(z) = P(Z \leq z) = P(0 \leq X \leq z) + P(1 \leq X \leq z + 1) =$$

$$= \int_0^z x dx + \int_1^{1+z} (2 - x) dx = \left[\frac{x^2}{2}\right]_0^z + \left[2x - \frac{x^2}{2}\right]_1^{1+z} =$$

$$= \frac{z^2}{2} - 0 + 2(1+z) - \frac{(1+z)^2}{2} - 2 + \frac{1}{2} = \frac{z^2}{2} + 2 + 2z - \frac{1}{2} - \frac{z^2}{2} - z - 2 + \frac{1}{2} = z.$$

QED[1]

[1] QED: Quod erat demonstrandum es una locución latina que significa 'lo que se quería demostrar' y se abrevia QED

## Generadores binarios de cambio de registro

Representemos por $b_i$ el bit i-ésimo.

Partiendo de una semilla $b_i = (a_1 b_{i-1} + a_2 b_{i-2} + \ldots + a_d b_{i-d})\ (2),\ con\ a_1, \ldots,\ a_d \in \{0,1\}$, se obtiene el bit i-ésimo a partir de una combinación lineal de los anteriores:

$$b_i = a_1 b_{i-1} + a_2 b_{i-2} + \cdots + a_d b_{i-d},\ \text{con}\ \ a_1, \ldots,\ a_d \in \{0,1\}.$$

Los ciclos serán de longitud $2^d$-1 a lo sumo (suponiendo $a_d$=1). Si se obtiene d veces consecutivas el valor 0, entonces el algoritmo degenera generando siempre este valor.

Usualmente se procede como sigue: se toma p>q y se hace

$$b_i = (b_{i-p} + b_{i-q}) \equiv 2.$$

El ciclo así generado tendrá una longitud de a lo sumo $2^p$-1 bits.

Si se quiere generar números pseudoaleatorios con precisión $2^{-L}$, se toma

$$u_i = \sum_{s=1}^{L} \frac{b_{s+iL}}{2^s}.$$

Con el fin de ahorrar esfuerzo de cálculo se suele tomar (método sugerido por Taustworthe)

$$u_i = \sum_{s=1}^{L} \frac{b_{s+it}}{2^s},$$

donde t se denomina decimación. Así se tiene que el período máximo se alcanza cuando se cumple $mcd(t,2^p\text{-}1)=1$. Los valores más habituales son

$$\begin{array}{lll} p = 98 & , & q = 27, \\ p = 521 & , & q = 32, \\ p = 607 & , & q = 273. \end{array}$$

## 5. Ejercicios

5.1. Genere números aleatorios entre 0 y 1 con los siguientes generadores congruenciales y determine el ciclo de vida de cada uno.

- xi+1 = (40xi + 13) mod33 x0 = 302
- xj+1 = (71xj + 57) mod341 x0 = 71
- xi+1= (71xi + 517) mod111 x0 = 171
- xi+1= (71561xi + 56822117) mod341157 x0 = 31767
- xi+1= (723xi + 531) mod314 x0 = 927
- xi+1 = (452xi + 37452) mod1231 x0 = 4571
- xi+1 = (17xi) mod37 x0 = 51
- xi+1 = (16xi + 4) mod14 x0 = 22

5.2. Genere 50 números entre 0 y 1 de 4 dígitos, mediante un generador de cuadrados medios cuya semilla sea
   a) 4567234902
   b) 3567345
   c) 1234500012

5.3. Genere números aleatorios con los siguientes generadores e identifique la longitud o ciclo de vida:
   a) Método de Lehmer. Semilla 87. Const. K=4
   b) Método de Lehmer. Semilla 69. Const. K=3
   c) Método de Lehmer. Semilla 99. Const. K=2
   - Cual de ellos es el de mayor longitud

5.4. Programe en una hoja de cálculo la serie congruencial X i+1 = (553 + 121 Xi) mod (177) con X0 = 23.
   a) Determine el ciclo de vida

5.5. Determine el período de los siguientes generadores congruenciales mixtos:
   - Xn+1 = (8 Xn+16) mod 100 y X0 = 15.
   - Xn+1 = (50 Xn+17) mod 64 y X0 = 13.
   - Xn+1 = (5 Xn+24) mod 32 y X0 = 7.
   - Xn+1 = (5 Xn+21) mod 100 y X0 = 3.
   - Xn+1 = (9 Xn+13) mod 32 y X0 = 8.

5.6. Determine el período de los siguientes generadores congruenciales multiplicativos:
   - Xn+1 = 203 Xn mod 105 y X0 = 17.
   - Xn+1 = 211 Xn mod 108 y X0 = 19.
   - Xn+1 = 221 Xn mod 103 y X0 = 3.
   - Xn+1 = 5Xn mod 64 y X0 = 7.
   - Xn+1 = 11 Xn mod 128 y X0 = 9.

5.7. Genere números aleatorios entre 0 y 1 con los siguientes generadores congruenciales y determine el ciclo de vida de cada uno.
   - Xn+1 = (40 Xn+13) mod 33 y X0 = 302.
   - Xn+1 = (71 Xn+57) mod 341 y X0 = 71.
   - Xn+1 = (71 Xn+517) mod 111 y X0 = 171.
   - Xn+1 = (71561 Xn+56822117) mod 341157 y X0 = 31767.
   - Xn+1 = (723 Xn+531) mod 314 y X0 = 927.

- Xn+1 = (452 Xn+37452) mod 1231 y X0 = 4571.
- Xn+1 = (17 Xn) mod 37 y X0 = 51.
- Xn+1 = (16Xn+4) mod 14 y X0 = 22.

5.8. Genere 50 números aleatorios entre 0 y 1 de 4 dígitos, mediante un generador de cuadrados medios cuya semilla sea:

- 4567234902. - 3567345 - · 1234500012