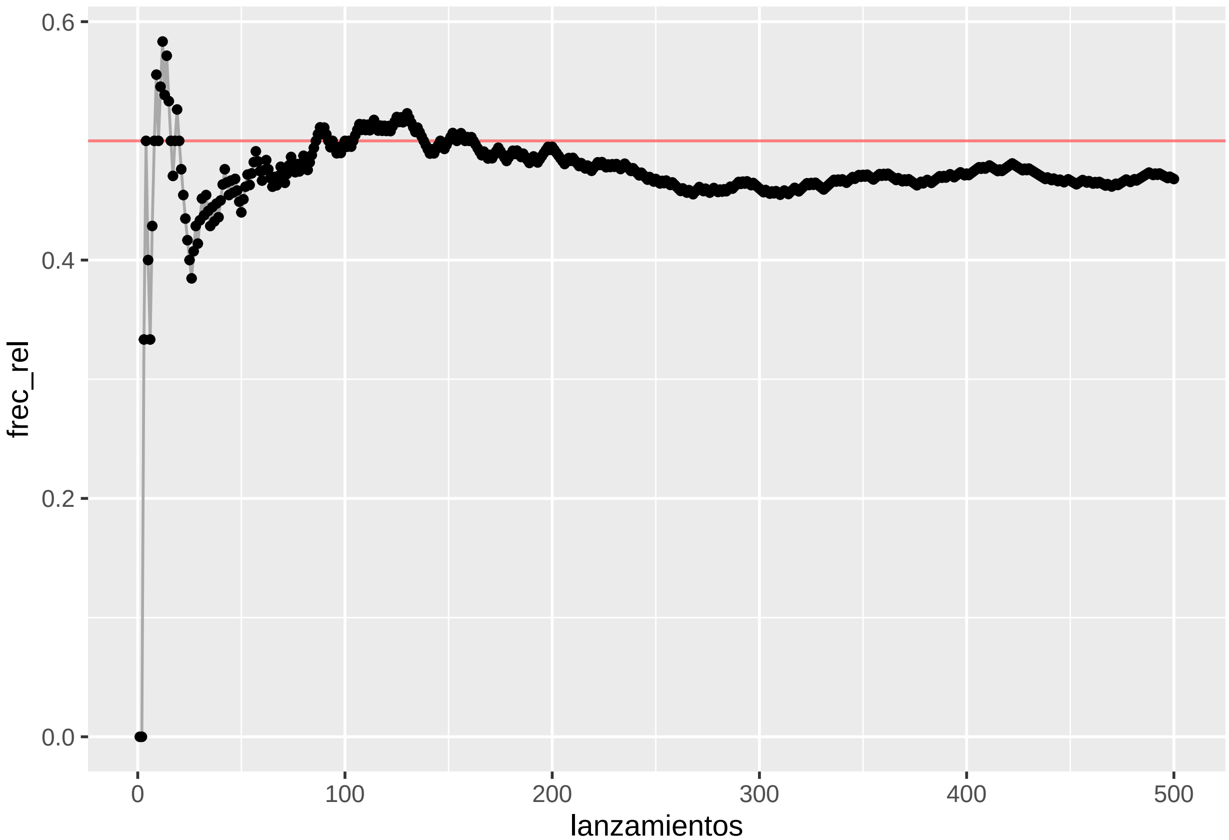
Tarea 1

Ana Muñiz, Jordi Legorreta, David López, Paulina Vásquez

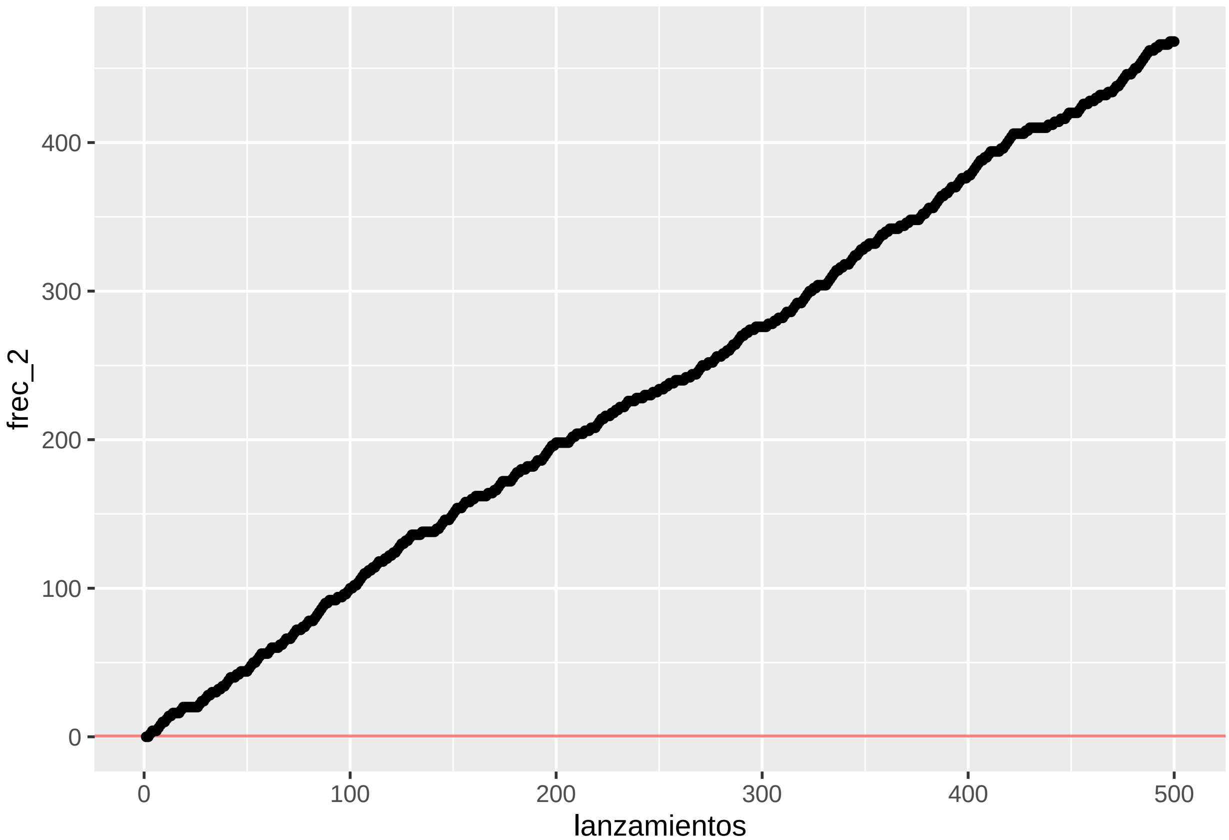
Ejercicio 1

En la variable **res**, guardamos si es águila o sol, en forma de 1 y 2. 1 águila y 2 sol. Utilizamos la función sample para reproducir las repeticiones, y creamos un contador de cuantas veces sale sol, que representa r.

```
library(ggplot2)
set.seed(164840)

#Forma manual
# n <- 500;r<- 0;res <- rep(0,500)
# plot(r/n, n)
# for(i in 1:n) {
#   res[i]<- sample(1:2, 1, replace = FALSE)
#   if (res[i] == 2)r <- r +1
#   points(r/i,i, col= "blue")
#   points(2*r-i, i, col= "red")}
# r<- 0;res <- rep(0,500)
# plot(2*r-n, n)
# for(i in 1:n) {
#   res[i]<- sample(1:2, 1, replace = FALSE)
#   if (res[i] == 2)r <- r +1
#   points(2*r-i, i, col= "red")}
# Alternativa
# prueba<- floor(runif(500, min=1, max=3))

n <- 500
lanzamientos <- sample(c("A", "S"), n, replace = TRUE)
frec_rel <- cumsum(lanzamientos == "A") / 1:n
frec_rel_df <- data.frame(lanzamientos = 1:n, frec_rel = frec_rel)

frec_2 <- 2*cumsum(lanzamientos == "A")
frec_2_df <- data.frame(lanzamientos = 1:n, frec_2 = frec_2)

ggplot(frec_rel_df, aes(x = lanzamientos, y = frec_rel)) +
  geom_hline(yintercept = 0.5, color = "red", alpha = 0.5) +
  geom_line(color = "darkgray") +
  geom_point(size = 1.2)
```




```r
ggplot(frec_2_df, aes(x = lanzamientos, y = frec_2)) +
  geom_hline(yintercept = 0.5, color = "red", alpha = 0.5) +
  geom_line(color = "darkgray") +
  geom_point(size = 1.2)
```



Ejercicio 2

Dar 5 ejemplos en donde se puede usar simulación.

1. Para simular como va a progresar el comportamiento de un virus en una determinada población. Simulación te va a permitir mover varibles (cantidad de doctores, hospitales, camas, etc) para ver que se puede esperar o que médidas se deben de aplicar.
2. En un negocio para optimizar el número de empleados y el tiempo de espera de los clientes. Por ejemplo, cuantas cajas tener abiertas en el super. Hasta se pueden hacer simulaciones haciendo por ejemplo una caja rápida.
3. Para generar más variables con un determinado comportamiento y hacer un estudio más robusto. Por ejemplo, si estas intentando ver el tiempo probmedio de un lugar a otro, no lo tienes que medir fisicamente, puedes a patir de un numero de mediciones, encontrar los parametros necesarios para entonces simular miles mas.
4. Se pueden simular eventos extraordinarios, como qué pasaría si por ejemplo hay una guerra que causa que caigan los precios del petroleo, se podría simular como van a reaccionar las instituciones financieras, y que consecuencias tendría en la bolsa.
5. Se puede simular una nueva ruta aerea, tomando en cuenta el tipo de avión, la distancia que se tiene que recorrer, la velocidad que alcanza el avion, etc.

Ejercicio 3

Una canoa con 3 mujeres y 3 hombres llega a una isla deshabitada, que información se necesita para modelar la sociedad.

- Primero necesitamos saber si las condiciones son favorables para que sobrevivan las personas, es decir, si hay alimento, agua potable y las condiciones geograficas lo permiten.
- Necesitamos saber el parentezco entre las personas que llegaron, para así saber la velocidad con la que se pueden reproducir estas personas.
- Necesitamos saber si el alimento es suficiente para que se reproduzcan de forma "descontrolada" o si se tienen que acatar a una restricción alimentaria.
- De las dos anteriores obtenemos una taza de reproduccion
- Necesitamos saber si estan bajo un riesgo externo, como si corren peligro por que un animal los quiera comer, lo cual significaría que en cualqueir momento se puede erradicar esa población
- Necesitamos saber si tienen medicamentos a su alance, para dar una mejor indicación de la tasa de supervivencia.

Ejercicio 4

Cómo se podría simular el modelo de una sala de sirugía bajo citas.

- Utilizando las probabilidades de la hora a la que llegaran los pacientes, simulamos como van a llegar en realidad los pacientes. Adicionalmente se toma en consideración la probabilidad de que no llegue.
- Utilizando las probabilidades de el tiempo que tarden durante la consulta simulamos cuanto tiempo se va a tardar en cada paciente.

Si pensamos el modelo como que llega un individuo a la vez, primero simulas a que hora va a llegar ese individuo, y luego simulas el tiempo que va a tardar dentro de la consulta, lo cual nos va a decir junto con la información de la hora de llegada del siguiente paciente, cuanto tiempo espero, y ahí se vuelve a simular el tiempo que va a tardar en consulta y así sucesivamente.

Lo que al final vamos a modelar es el tiempo que espera cada paciente, indicando con negativos si llega tarde y el doctor es el que tiene que esperar.

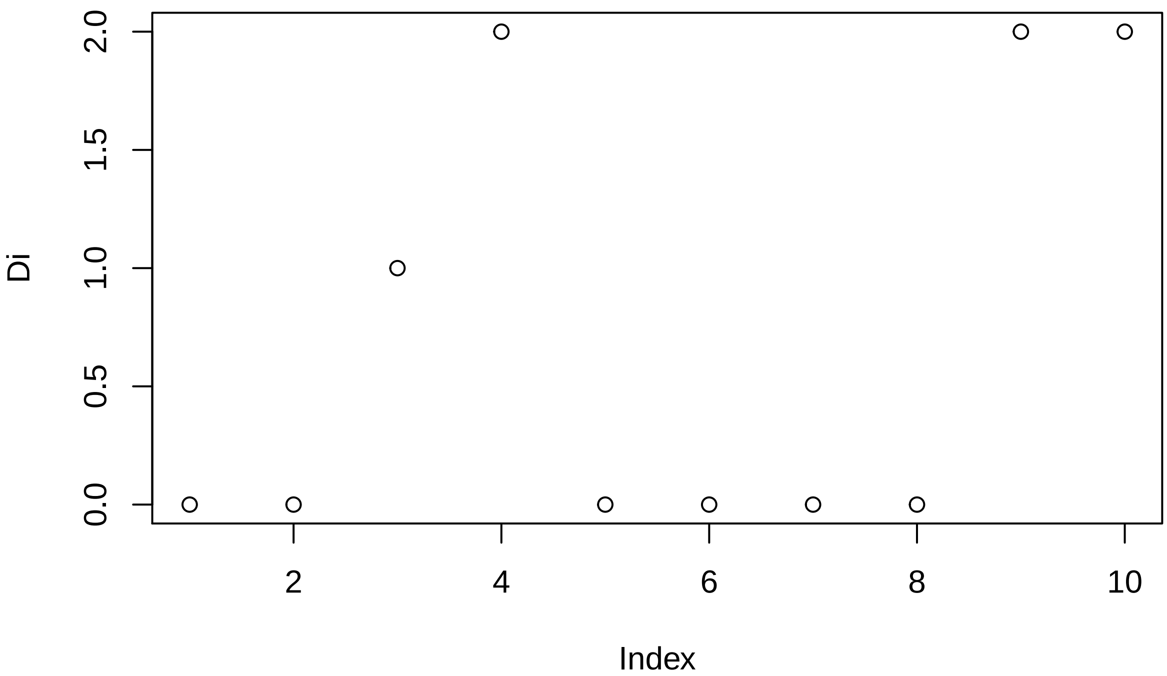
ti = Tiempo de llegada del i-ésimo paciente Ti = ti - ti-1 = tiempo entre las llegadas del i- 1 e i paciente (tiempo de interarribo). Si = tiempo de servicio al paciente i. Di = tiempo de espera en cola del paciente i. ci = ti + Di + Si = tiempo transcurrido entre la llegada del paciente i y su salida del sistema. ei = tiempo de ocurrencia del i-ésimo evento de cualquier tipo (reloj de simulación).

```
ti <- c(-2,-1,0,1,2)
prob_ti <- c(1/10, 1/5, 2/5, 1/5, 1/10)
datafr <- cbind(ti,prob_ti)
Ti <-sample(ti, size = 10, replace = TRUE, prob = prob_ti)

#Duracion
Si_2 <-c(2,3,4,5,6,7,8,9)
prob_Si <-c(1/10,1/10,1/10,1/5,1/5,1/10,1/10,1/10)
Si <-sample(Si_2, size = 10, replace = TRUE, prob = prob_Si)

n <- 10
inicio <- rep(0,n)
termino<- rep(0,n)
Di<- rep(0,n)

if(Ti[1]<=0){
  termino[1]<- Si[1]
  Di[1]<- 0
}else{
  termino[1]<- Si[1] + Ti[1]
  Di[1]<- Ti[1]
```

```r
}
for (i in 2:n)
{
 # termino[i]<- termino[i-1]+ Di[i]

  if(Ti[i]<=0){
    inicio[i]<- termino[i-1]
     termino[i]<- Si[i] + inicio[i]

  }else{
    inicio[i]<- termino[i-1] + Ti[i]
    termino[i]<- Si[i] + inicio[i]
  }
  Di[i]<- inicio[i]-termino[i-1]
  #inicio[i]-termino[i-1]

}
#cuadro con los datos
datafr <- cbind(Ti, inicio, Si, termino,Di)
datafr
```

```
##        Ti inicio Si termino Di
##  [1,] -1      0  8       8  0
##  [2,] -1      8  8      16  0
##  [3,]  1     17  9      26  1
##  [4,]  2     28  5      33  2
##  [5,] -1     33  9      42  0
##  [6,]  0     42  4      46  0
##  [7,] -1     46  2      48  0
##  [8,] -2     48  2      50  0
##  [9,]  2     52  5      57  2
## [10,]  2     59  6      65  2
```

```r
#grafica del tiempo de espera
plot(Di)
```



Ejercicio 5

Calcular el periodo del GLC $Z_i \equiv (5Z_{i-1} + 3) \mod 31$

```
glc = function(m=31,a=5,c=3,z0){
  z = NULL
  z[1] = z0
  for(i in 2:16){
    z[i] = (a*z[i-1]+c) %% m
  }
  z
}
for(i in 1:15){
  print(glc(z0=i))
}
```

```
##  [1]  1  8 12  1  8 12  1  8 12  1  8 12  1  8 12  1
##  [1]  2 13  6  2 13  6  2 13  6  2 13  6  2 13  6  2
##  [1]  3 18  0  3 18  0  3 18  0  3 18  0  3 18  0  3
##  [1]  4 23 25  4 23 25  4 23 25  4 23 25  4 23 25  4
##  [1]  5 28 19  5 28 19  5 28 19  5 28 19  5 28 19  5
##  [1]  6  2 13  6  2 13  6  2 13  6  2 13  6  2 13  6
##  [1] 7 7 7 7 7 7 7 7 7 7 7 7 7 7 7 7
##  [1]  8 12  1  8 12  1  8 12  1  8 12  1  8 12  1  8
##  [1]  9 17 26  9 17 26  9 17 26  9 17 26  9 17 26  9
##  [1] 10 22 20 10 22 20 10 22 20 10 22 20 10 22 20 10
##  [1] 11 27 14 11 27 14 11 27 14 11 27 14 11 27 14 11
##  [1] 12  1  8 12  1  8 12  1  8 12  1  8 12  1  8 12
##  [1] 13  6  2 13  6  2 13  6  2 13  6  2 13  6  2 13
##  [1] 14 11 27 14 11 27 14 11 27 14 11 27 14 11 27 14
##  [1] 15 16 21 15 16 21 15 16 21 15 16 21 15 16 21 15
```

Podemos observar que el período del GLC es 3, aunque hay una excepción en $z_0 = 7$, ya que el período en este caso es 1.

Ejercicio 6

6) Mostrar que el promedio de las U_i's tomados de un ciclo completo de un GLC de periodo completo es $\frac{1}{2} - \frac{1}{2m}$.

Demostración:

Un generador lineal congruencial se define como sigue:

$Z_i \equiv (a Z_{i-1} + c) \mod m$; Z_0 = valor inicial

m = módulo

$0 \leq Z_i \leq m-1 \quad \forall i$; a = multiplicador

$U_i = \frac{Z_i}{m}$ (Valor de cada U_i) ; c = incremento

En este caso, el GLC es de periodo completo.

De esta manera, cada valor del de la congruencia debe aparecer al menos, una vez. Entonces, del promedio de todos los valores (Z_i) se obtiene que

$$\frac{Z_0 + \dots + Z_{m-1}}{m} = \frac{(m-1)}{2}$$

ya que $Z_i = 0$ para alguna $i \in \{0, \dots, m-1\}$

$$\frac{Z_0}{m} + \dots + \frac{Z_{m-1}}{m} = \frac{m}{2} - \frac{1}{2} \iff U_0 + \dots + U_{m-1} = \frac{m}{2} - \frac{1}{2}$$

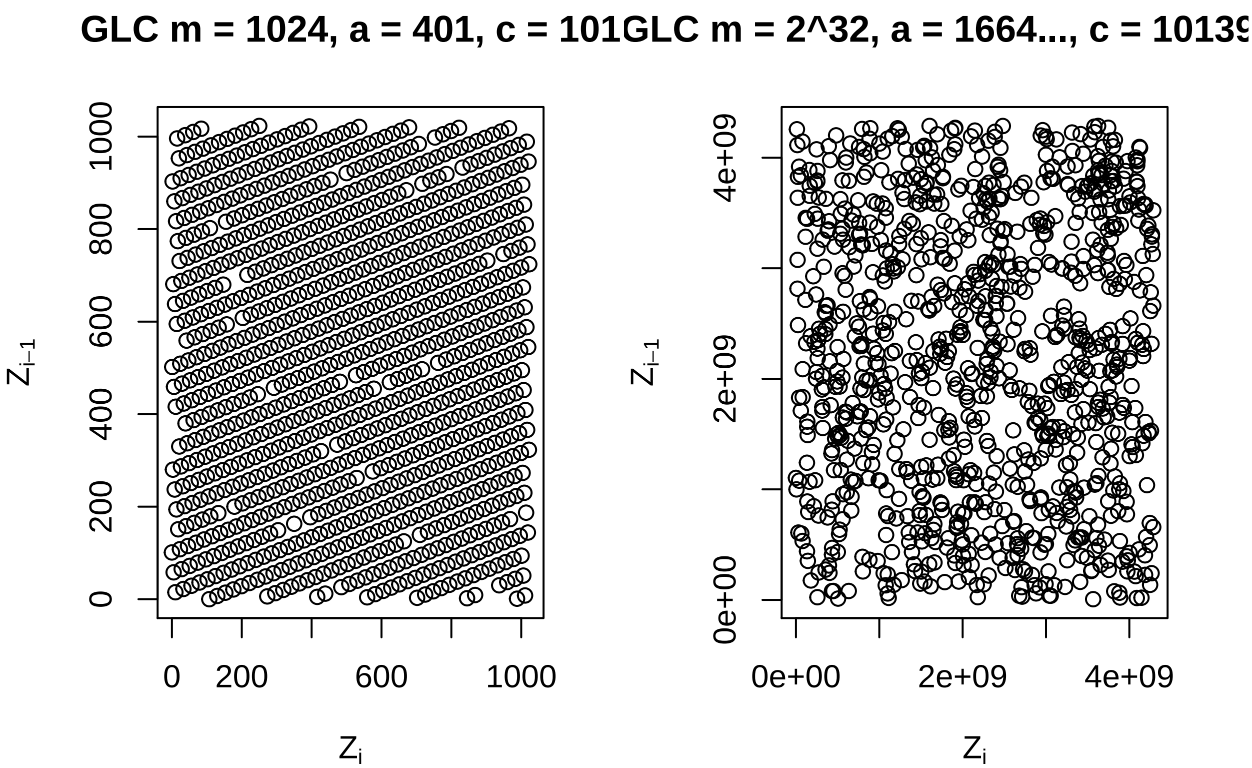
$$\iff \frac{U_0 + \dots + U_{m-1}}{m} = \frac{m}{2m} - \frac{1}{2m} \iff \frac{U_0 + \dots + U_{m-1}}{m} = \frac{1}{2} - \frac{1}{2m}$$

Ejercicio 7

Hacer una gráfica de dispersión de puntos de (Z_i, Z_{i+1}) para $i = 1, ..., n-1$ para el GLC con parámetros $m = 1024, a = 401, c = 101$ y para el GLC $m = 2^{32}, a = 1,664,525, c = 1,013,904,223$

```
lgc = function(z0,a,c,m){
  z = NULL
  z[1] = z0
```

```r
  for(i in 2:1000){
    z[i] = (a*z[i-1]+c) %% m
  }
  z
}
z1024 = lgc(z0=23,a=401,c=101,m=1024)
z2_32 = lgc(z0=54,a=1664525,c=1013904223,m=2^32)

par(mfrow = c(1,2))
plot(z1024[1:999],z1024[2:1000],xlab=expression(Z[i]),
     ylab=expression(Z[i-1]),main="GLC m = 1024, a = 401, c = 101")
plot(z2_32[1:999],z2_32[2:1000],xlab=expression(Z[i]),
     ylab=expression(Z[i-1]),main="GLC m = 2^32, a = 1664..., c = 1013904...")
```



Ejercicio 8

8) Probar que la parte fraccional de la suma de uniformes [0,1] $U_1+U_2+\cdots+U_k$ es también uniforme en el intervalo [0,1].

Demostración:

Se define $(x) = x - \lfloor x \rfloor$ a la parte fraccional de x. Si probamos que dados $U_1, U_2 \sim$ Uniforme(0,1), de manera inductiva se puede probar el resultado general, puesto que para $U_i \sim$ Uniforme(0,1) $i \in \{1, \dots, k\}$ iid se tiene que

$$(U_1+U_2+\dots+U_k) = ((U_1+U_2+\dots+U_{k-1})+U_k) = (U+U_k)$$

$$(U := (U_1+\dots+U_{k-1}))$$

Todo resulta en probar el caso $k=2$, la base de inducción:

Sean $U_1, U_2 \sim$ Uniforme(0,1) iid y sea $S_2 = U_1+U_2$ con f.d.p.

$$f_{S_2}(s) = \begin{cases} s & 0 \le s < 1 \\ 2-s & 1 \le s \le 2 \\ 0 & c.o.c. \end{cases}$$

Si $s \in [0,1] \Rightarrow \mathbb{P}[(S_2) \le s]$ tiene dos casos:

Caso I $\quad 0 \le S_2 < 1 \Rightarrow (S_2) \le s \iff S_2 \le s$

Caso II $\quad 1 \le S_2 \le 2 \Rightarrow (S_2) = S_2 - 1 \Rightarrow 0 \le S_2 - 1 \le 1 \Rightarrow S_2 - 1 \le 1+s$

Si integramos para obtener la f.p.a. resulta que

$$F_{(S_2)}(s) = \mathbb{P}[(S_2) \le s] = \int_{u=0}^{s} f_{S_2}(u)\,du + \int_{u=1}^{1+s} f_{S_2}(u)\,du = \int_0^s u\,du + \int_1^{1+s} (2-u)\,du$$

$$= \frac{u^2}{2}\Big]_0^s + \left(2u - \frac{u^2}{2}\right)\Big]_1^{1+s} = \frac{s^2}{2} + 2(1+s) - \frac{(1+s)^2}{2} - \left[2 - \frac{1}{2}\right]$$

$$= \cancel{\frac{s^2}{2}} + \cancel{2} + 2s - \cancel{\frac{1}{2}} - \cancel{\frac{s^2}{2}} - s - \cancel{2} + \cancel{\frac{1}{2}} = s$$

Por lo tanto, $F_{(S_2)}(s) = s \; \mathbb{1}_{(0,1)}(s)$

Supongamos que $(U_1 + U_2 + \dots + U_k) \sim$ Uniforme $(0,1)$ para cierta $k \in \mathbb{N}$. Veamos si se cumple $\forall\, k \geq 2$

$(U_1 + \dots + U_k + U_{k+1}) = ((U_1 + \dots + U_k) + U_{k+1})$

Sea $U = (U_1 + \dots + U_k)$ y, por hipótesis, $U \sim$ Uniforme$(0,1)$

$(U + U_{k+1}) = U_0 \sim$ Uniforme $(0,1)$ por la base de inducción.

Por lo tanto, la parte fraccional de la suma de uniformes $[0,1]$ se distribuye uniforme $[0,1]$.

Ejercicio 9

Primero definimos una función que genere los primeros n dígitos de la secuencia de Fibonacci modulo 5.

```
genFibonacci <- function(x0,x1,m,n)
{
  aux<-c(x0,x1)
  for (i in 1:n)aux<-c(aux,((aux[i]+aux[i+1]) %% m))
  return(aux)
}
```

De probar con diferentes combinaciones de números, concluímos que hay dos ciclos diferentes:

El primero, observado con semillas como $x_0 = 0$ y $x_1 = 1$, y $x_0 = 6$ y $x_1 = 7$ es la siguiente secuencia de **20** dígitos

```
genFibonacci(0,1,5,18)
```

```
##  [1] 0 1 1 2 3 0 3 3 1 4 0 4 4 3 2 0 2 2 4 1
```

El segundo ciclo, observado para semillas como $x_0 = 3$ y $x_1 = 4$, y $x_0 = 8$ y $x_1 = 9$ es la siguiente secuencia de **4** dígitos

```
genFibonacci(3,4,5,2)
```

```
## [1] 3 4 2 1
```

Ejercicio 10

Definimos una función que genere el siguiente número cuadrado medio de Neumann.

```
cmNeumann<-function(z0)
{
  aux<-toString(z0^2)
  if (nchar(aux)<4)
  {
    aux<-paste(0,0,0,aux, sep="")
    aux<-substr(aux, nchar(aux)-3, nchar(aux))
  }
  return(as.numeric(substring(aux,2,3)))
}
```

Ahora, hacemos un programa que identifique los bucles en la generación de números aleatorios para semillas entre 0 y 99.

```
n <- 99
a <- matrix(, nrow = 65, ncol = 15)
j <- 1;i <- 1;k <- 1
a[i,j] <- 0

while(k < n + 1)
{
  aux <- cmNeumann(a[j,i])
  if(aux == a[j,i] || (i>2 && aux == a[j,i-1])) #Bucle de periodo 1 o 2
  {
    j <- j + 1
    i <- 1
    k <- j - 1
```

```
    while(k %in% a)
    {
      k <- k + 1
    }
    if(j < nrow(a) + 1)a[j,i] <- k
  }
  else
  {
    i <- i + 1
    a[j,i] <- aux
  }
}
```

Table 1: Secuencias del Generador de Cuadrados Medios de Neumann

0														
1	0													
2	0													
3	0													
4	1	0												
5	2	0												
6	3	0												
7	4	1	0											
8	6	3	0											
9	8	6	3	0										
10														
11	12	14	19	36	29	84	5	2	0					
13	16	25	62	84	5	2	0							
15	22	48	30	90	10									
17	28	78	8	6	3	0								
18	32	2	0											
20	40	60												
21	44	93	64	9	8	6	3	0						
23	52	70	90	10										
24	57	24												
26	67	48	30	90	10									
27	72	18	32	2	0									
31	96	21	44	93	64	9	8	6	3	0				
33	8	6	3	0										
34	15	22	48	30	90	10								
35	22	48	30	90	10									
37	36	29	84	5	2	0								
38	44	93	64	9	8	6	3	0						
39	52	70	90	10										
41	68	62	84	5	2	0								
42	76	77	92	46	11	12	14	19	36	29	84	5	2	0
43	84	5	2	0										
45	2	0												
47	20	40	60											
49	40	60												
50														
51	60													
53	80	40	60											

Table 1: Secuencias del Generador de Cuadrados Medios de Neumann

54	91	28	78	8	6	3	0							
55	2	0												
56	13	16	25	62	84	5	2	0						
58	36	29	84	5	2	0								
59	48	30	90	10										
61	72	18	32	2	0									
63	96	21	44	93	64	9	8	6	3	0				
65	22	48	30	90	10									
66	35	22	48	30	90	10								
69	76	77	92	46	11	12	14	19	36	29	84	5	2	0
71	4	1	0											
73	32	2	0											
74	47	20	40	60										
75	62	84	5	2	0									
79	24	57												
81	56	13	16	25	62	84	5	2	0					
82	72	18	32	2	0									
83	88	74	47	20	40	60								
85	22	48	30	90	10									
86	39	52	70	90	10									
87	56	13	16	25	62	84	5	2	0					
89	92	46	11	12	14	19	36	29	84	5	2	0		
94	83	88	74	47	20	40	60							
95	2	0												
97	40	60												
98	60													
99	80	40	60											

Nota: El último número en cada secuencia es el inicio de un bucle

Comentarios sobre el código:

Variables:

- n: Número máximo evaluado en la función de cuadrados medios de Neumann
- a: Matriz que almacena las secuencias y bucles
- j: Contador de renglones
- i: Contador de columnas
- k: Número con el que iniciará la siguiente secuencia
- El programa funciona gracias a un *while* que corre mientras todavia hayan secuencias posibles a generar.
- Calculamos el `cmNeumann` de la entrada correspondiente al (i,j) en cuestión:
  - Si el `cmNeumann` es igual a la semilla, significa que hay un bucle de periodo 1.
  - Si el `cmNeumann` es igual a la entrada anterior a la semilla, significa que hay un bucle 2.
- En cada renglón de la matriz hay una secuencia.

Ejercicio 11

Empezamos por definir una función que a partir de un *string*, almacene por separado los caracteres que lo componen.

```
digits <- function(digitos)
{
  aux<-numeric(0)
  for (i in 1:nchar(digitos))aux<-c(aux,as.numeric(substring(digitos, i, i)))
  return(aux)
}
```

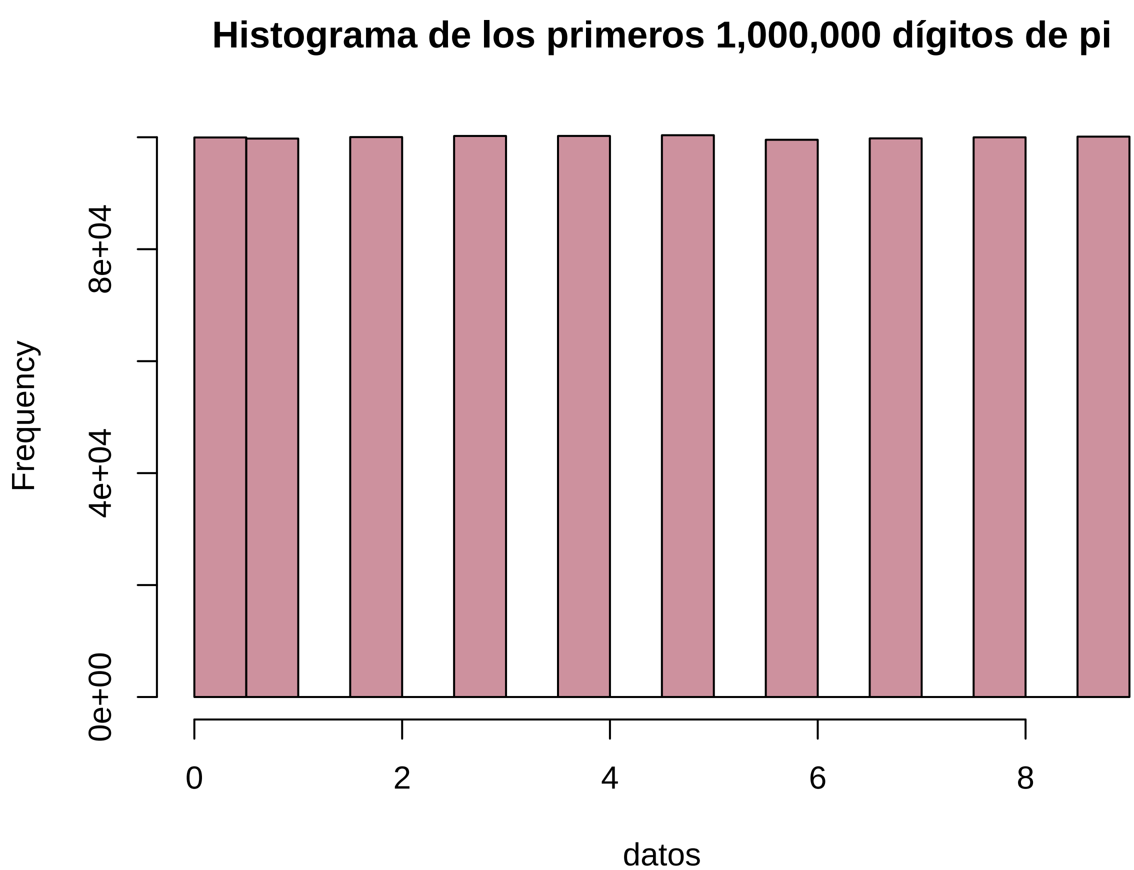
Ahora leemos el archivo *pi.txt* con el primer millón de dígitos de π; con ayuda de la funcion de definimos procesamos los datos para poder usarlos en pruebas de uniformidad e independencia. La razón por la cual trabajamos con *strings* (en lugar de *int*) es para evitar un `intOverflow`.

```
fileName <- "pi.txt"
w<-readChar(fileName, file.info(fileName)$size)
```

```
#w<-substring(w, 1, 100000)
datos<-digits(w)
```

Hacemos un histograma de los datos y vemos que parecen distribuirse de forma uniforme discreta, tal como esperábamos.

```
#Histograma
hist(datos, col='pink3',main = "Histograma de los primeros 1,000,000 dígitos de pi")
```



Ejercicio 12

12. sea X_1 : cara que cae en dado 1

X_2 : cara que cae en dado 2

Si para el dado 1 el valor 1 aparece el doble de veces

$\Rightarrow$ la probabilidad de que salga 1 es el doble que para cualquier otro número, así tenemos que:

$$\mathbb{P}(X_1 = 1) = 2x \qquad \mathbb{P}(X_1 = 4) = x$$
$$\mathbb{P}(X_1 = 2) = x \qquad \mathbb{P}(X_1 = 5) = x$$
$$\mathbb{P}(X_1 = 3) = x \qquad \mathbb{P}(X_1 = 6) = x$$

como $\sum_{i=1}^{6} \mathbb{P}(X_1 = i) = 7x = 1 \quad \Rightarrow \quad x = \frac{1}{7}$

para el dado 2 ocurre lo mismo para el valor de 6, $\therefore$

$$\mathbb{P}(X_2 = 1) = 1/7 \qquad \mathbb{P}(X_4 = 4) = 1/7$$
$$\mathbb{P}(X_2 = 2) = 1/7 \qquad \mathbb{P}(X_5 = 5) = 1/7$$
$$\mathbb{P}(X_3 = 3) = 1/7 \qquad \mathbb{P}(X_6 = 6) = 2/7$$

Así,

$$
\begin{aligned}
p_s &= \mathbb{P}(X_1 + X_2 = s) \\
&= \mathbb{P}(X_2 = s - X_1) \\
&= \mathbb{P}(X_2 = s - X_1 \mid X_1 = 1)\,\mathbb{P}(X_1 = 1) + \dots + \mathbb{P}(X_2 = s - X_1 \mid X_1 = 6)\,\mathbb{P}(X_1 = 6) \\
&= \sum_{i=1}^{6} \mathbb{P}(X_2 = s - i \mid X_1 = i)\,\mathbb{P}(X_1 = i) \\
&= \sum_{2}^{6} \frac{1}{7}\mathbb{P}(X_2 = s - i \mid X_1 = i) + \frac{2}{7}\mathbb{P}(X_2 = s - 1 \mid X_1 = 1) \\
&= \sum_{2}^{6} \frac{1}{7}\mathbb{P}(X_2 = s - i) + \frac{2}{7}\mathbb{P}(X_2 = s - 1)
\end{aligned}
$$

// por ley de prob. total

Hay indep entre $X_2 = s - i$ y $X_i = i$

con $\mathbb{P}(X_2 = i) = \begin{cases} \frac{1}{7} & i = 1, \dots, 5 \\ \frac{2}{7} & i = 6 \\ 0 & \text{c.o.c.} \end{cases}$