Universidad del Valle de Guatemala
Facultad de Ingeniería
Departamento de Ciencias de la Computación

Mini-Proyecto #2

Link del repositorio: https://github.com/alegudiel/Modelacion_MiniProyecto_2

Alejandra Gudiel 19232
Diego Álvarez 19498

Ejercicio 1

El triángulo de Sierpinski es uno de los fractales más famosos. Un fractal se puede construir a través de saltos aleatorios entre funciones determinísticas (juegos de caos).

Considere las siguientes tres funciones:

- Point f1(Point p): return Point(p.x/2, p.y/2)
- Point f2(Point p): return Point(p.x/2 + 0.5, p.y/2)
- Point f3(Point p): return Point(p.x/2 + 0.25, p.y/2 + 0.5)

Estas funciones toman como parámetro el punto anterior en el fractal, y lo utilizan para generar siguiente punto (paso determinístico).

La idea de los juegos del caos es simular una variable aleatoria X, para elegir la función a utilizar para calcular el siguiente punto. Noten que X debe ser discreta por tanto: P(X=x1) =p1, P(X=x2) =p2, P(X=x3) =p3, tal que p1 +p2 +p3 = 1

Tasks:

1. Cree un programa que simule 100,000 veces X para elegir entre f1, f2 y f3, dibuje un triángulo de Sierpinski

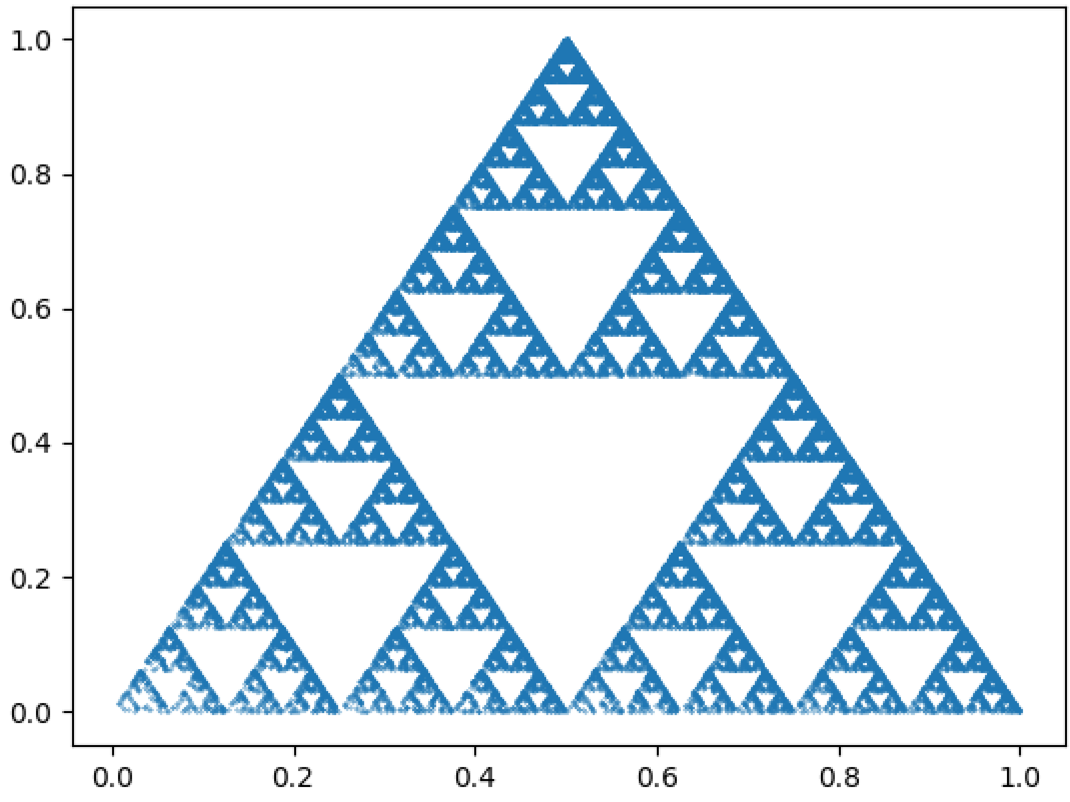


Imagen #1: triángulo de Sierpinski simulando 100,000 veces X

2. Determine experimentalmente p1, p2, p3 que hacen su dibujo más denso

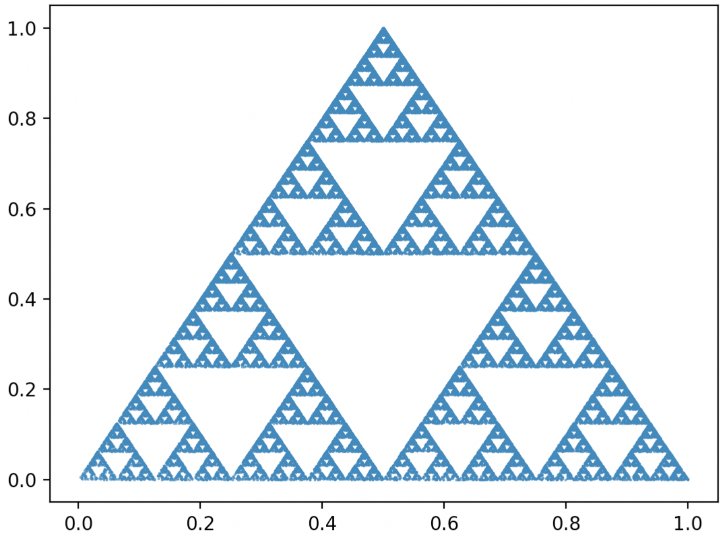


Imagen #2: Triángulo de Sierpinski con p1=0.25, p2=0.55, p3=0.20

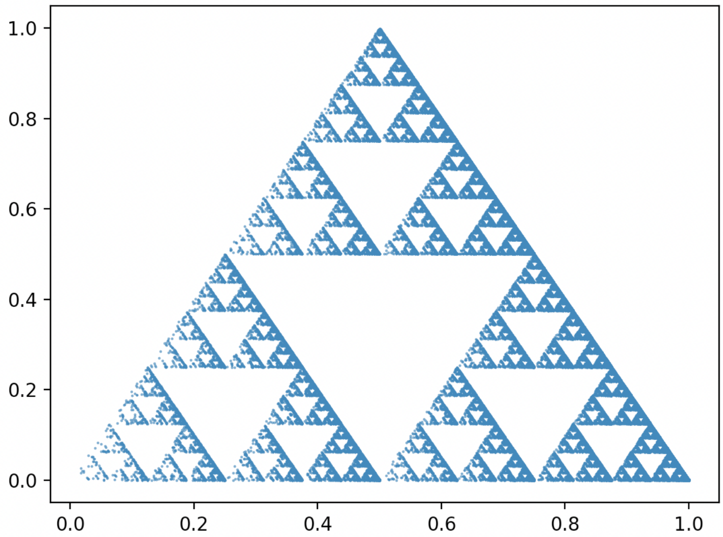


Imagen #3: Triángulo de Sierpinski con p1=0.15, p2=0.67, p3=0.17

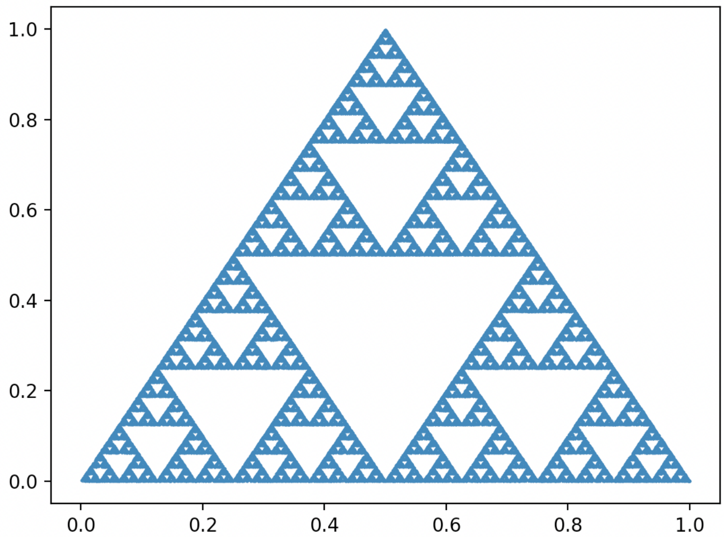


Imagen #4: Triángulo de Sierpinski con p1=0.33, p2=0.66, p3=0.10

Ejercicio 2

En general, los juegos del caos pueden definirse como un Framework(F,X,n), donde F es conjunto de funciones determinísticas, y X es una variable aleatoria discreta con distribución probabilística P = {pi | P(X=i) = pi}, y n es la cantidad de puntos a dibujar. Noten que |P| = k.
Para este fractal en específico considere:

F = f1(x, y) (x*0.85 + y*0.04 + 0.0, x*-0.04 + y*0.85 + 1.6)
f2(x, y) (-0.15*x + 0.28*y + 0.0, x*0.26 + y*0.24 + 0.44)
f3(x, y) (x*0.2 + y*-0.26 + 0.0, x*0.23 + y*0.22 + 1.6)
f4(x, y) (x*0.0 + y*0.0, x*0.0 + y*0.16)
P = {0.85, 0.07, 0.07, 0.01}
n = 100000

Tasks:

1. Cree un programa que corra el anterior juego del caos y muestre el dibujo resultante

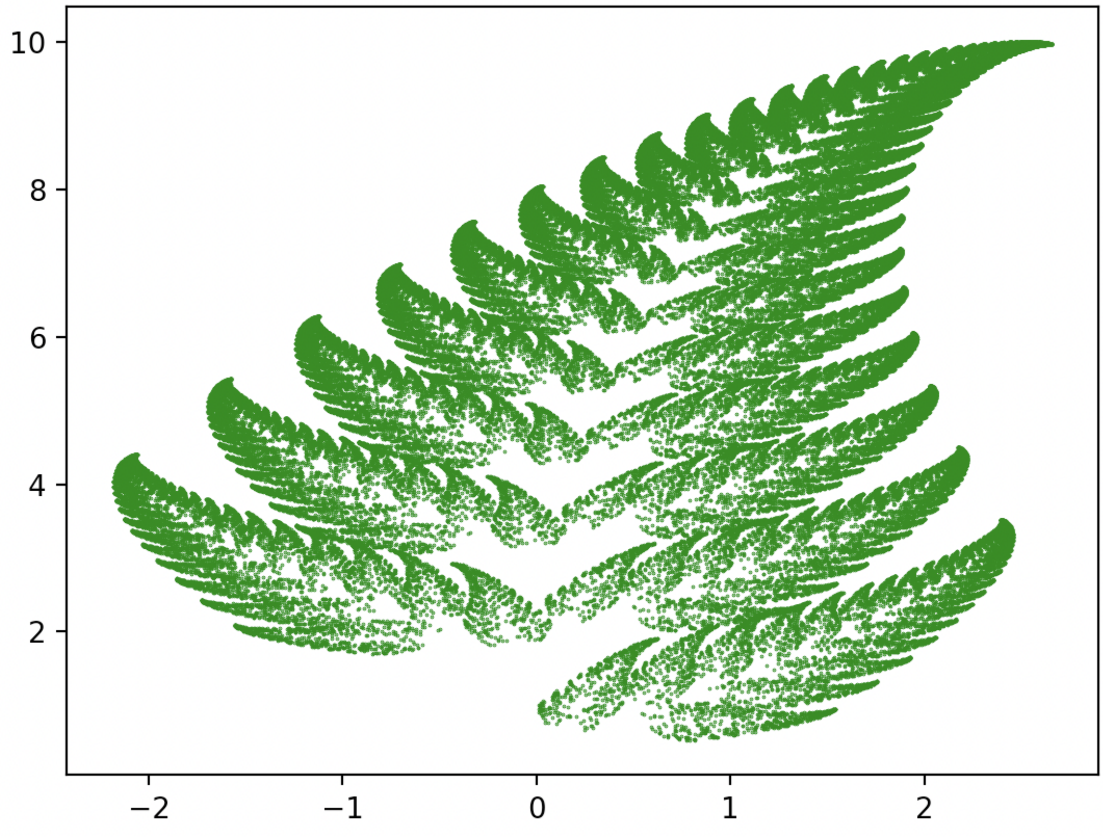


Imagen #5: dibujo resultante

Ejercicio 3

Considere las siguientes dos funciones generados de pseudo randoms:

Generador 1:

$$x_n = 5^5 x_{n-1} \bmod (2^{35} - 1)$$

Generador 2:

$$x_n = 7^5 x_{n-1} \bmod (2^{31} - 1)$$

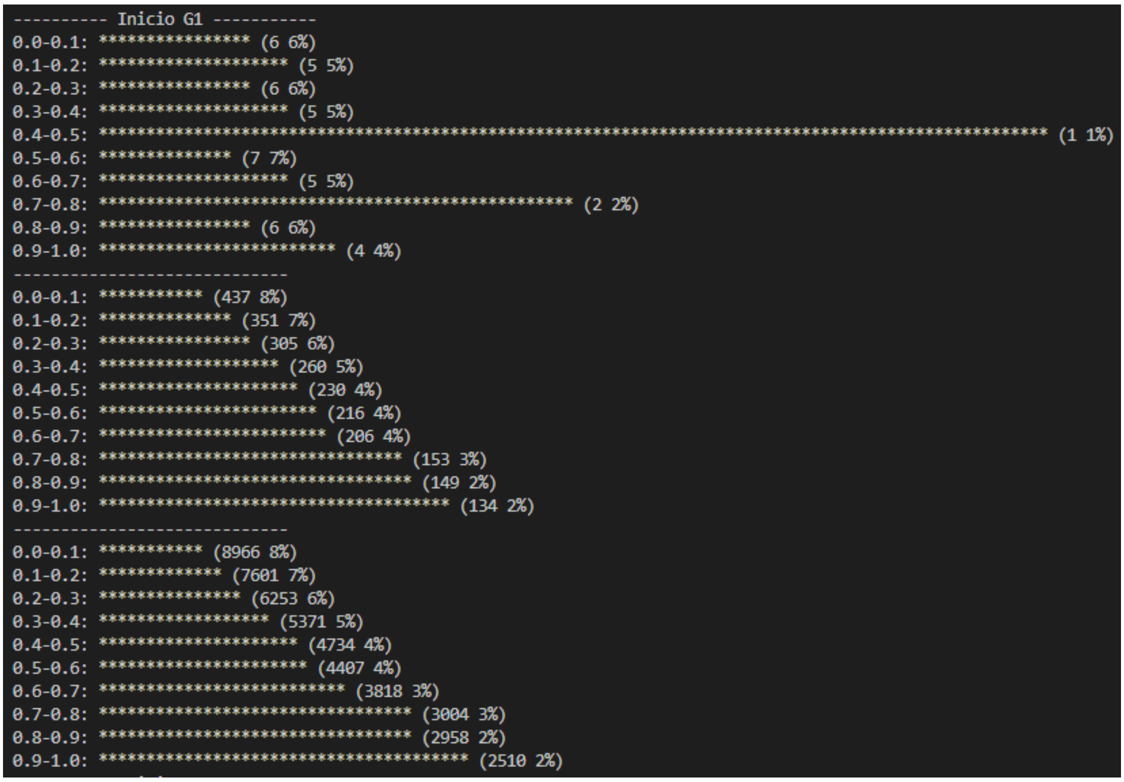
Y considere al generador de números aleatorios uniformes en (0, 1) default de su lenguaje de programación de elección, como un tercer generador de números aleatorios.

Generador 3: (baseline)

$$x_n = Math.random()$$

Tasks:

1. Construya un programa que compare estos tres generadores a través del histograma asteriscos (de 0 a 1 con saltos de 0.1) Use tres comparaciones para 100, 5,000 y 100,000 repeticiones.



```
---------- Inicio G1 -----------
0.0-0.1: **************** (6 6%)
0.1-0.2: ******************** (5 5%)
0.2-0.3: **************** (6 6%)
0.3-0.4: ******************** (5 5%)
0.4-0.5: **************************************************************************************************** (1 1%)
0.5-0.6: ************** (7 7%)
0.6-0.7: ******************** (5 5%)
0.7-0.8: ************************************************** (2 2%)
0.8-0.9: **************** (6 6%)
0.9-1.0: ************************* (4 4%)
-----------------------------
0.0-0.1: *********** (437 8%)
0.1-0.2: ************** (351 7%)
0.2-0.3: **************** (305 6%)
0.3-0.4: ******************* (260 5%)
0.4-0.5: ********************* (230 4%)
0.5-0.6: *********************** (216 4%)
0.6-0.7: ************************ (206 4%)
0.7-0.8: ******************************** (153 3%)
0.8-0.9: ********************************* (149 2%)
0.9-1.0: ************************************* (134 2%)
-----------------------------
0.0-0.1: *********** (8966 8%)
0.1-0.2: ************* (7601 7%)
0.2-0.3: *************** (6253 6%)
0.3-0.4: ****************** (5371 5%)
0.4-0.5: ********************* (4734 4%)
0.5-0.6: ********************** (4407 4%)
0.6-0.7: ************************** (3818 3%)
0.7-0.8: ********************************* (3004 3%)
0.8-0.9: ********************************* (2958 2%)
0.9-1.0: *************************************** (2510 2%)
```

Imagen #6: generador 1

```
---------- Inicio G2 -----------
0.0-0.1: *********** (9 9%)
0.1-0.2: ************** (7 7%)
0.2-0.3: ************** (7 7%)
0.3-0.4: ********************************* (3 3%)
0.4-0.5: **************** (6 6%)
0.5-0.6: ************ (8 8%)
0.6-0.7: **************** (6 6%)
0.7-0.8: ************************************************** (2 2%)
0.8-0.9: ************************* (4 4%)
0.9-1.0:  (0 0%)
-----------------------------
0.0-0.1: ********** (465 9%)
0.1-0.2: ************* (382 7%)
0.2-0.3: *************** (315 6%)
0.3-0.4: ******************* (254 5%)
0.4-0.5: ******************* (255 5%)
0.5-0.6: ************************** (192 3%)
0.6-0.7: **************************** (178 3%)
0.7-0.8: ******************************** (155 3%)
0.8-0.9: ******************************** (156 3%)
0.9-1.0: **************************************** (123 2%)
-----------------------------
0.0-0.1: ********** (9147 9%)
0.1-0.2: ************* (7574 7%)
0.2-0.3: *************** (6398 6%)
0.3-0.4: ****************** (5410 5%)
0.4-0.5: ******************** (4801 4%)
0.5-0.6: *********************** (4173 4%)
0.6-0.7: *************************** (3696 3%)
0.7-0.8: ***************************** (3378 3%)
0.8-0.9: ********************************* (3001 3%)
0.9-1.0: ************************************* (2691 2%)
```

Imagen #7: generador 2

```
---------- Inicio G3 -----------
0.0-0.1: *********** (9 9%)
0.1-0.2: ******** (12 12%)
0.2-0.3: ********* (11 11%)
0.3-0.4: ******* (13 13%)
0.4-0.5: *********** (9 9%)
0.5-0.6: ************ (8 8%)
0.6-0.7: ************** (7 7%)
0.7-0.8: ************** (7 7%)
0.8-0.9: ******* (13 13%)
0.9-1.0: ********* (11 11%)
-----------------------------
```

```
-----------------------------
0.0-0.1: ********** (498 9%)
0.1-0.2: ********* (522 10%)
0.2-0.3: *********** (451 9%)
0.3-0.4: ********* (514 10%)
0.4-0.5: ********* (518 10%)
0.5-0.6: ********* (504 10%)
0.6-0.7: ********* (505 10%)
0.7-0.8: ********** (465 9%)
0.8-0.9: ********* (508 10%)
0.9-1.0: ********* (515 10%)
-----------------------------
```

```
-----------------------------
0.0-0.1: ********* (10055 10%)
0.1-0.2: ********* (10101 10%)
0.2-0.3: ********** (9887 9%)
0.3-0.4: ********** (9972 9%)
0.4-0.5: ********* (10069 10%)
0.5-0.6: ********* (10024 10%)
0.6-0.7: ********** (9809 9%)
0.7-0.8: ********** (9859 9%)
0.8-0.9: ********* (10079 10%)
0.9-1.0: ********* (10145 10%)
-----------------------------
```

Imagen #8: generador 3

2. ¿Qué generador le parece mejor? (considere solamente Generador 1 y Generador 2) ¿Por qué?

   Nos parece mejor el generador 2 ya que, hay una mayor variabilidad en los porcentajes de cada rango, o sea la probabilidad de obtener un valor en ese rango. Por otro lado, el generador 2 tiende a ser más proporcional a comparación del generador 1, de este modo se asemeja a una distribución normal.

```
0.0-0.1: ********** (9147 9%)
0.1-0.2: ************* (7574 7%)
0.2-0.3: *************** (6398 6%)
0.3-0.4: ****************** (5410 5%)
0.4-0.5: ******************** (4801 4%)
0.5-0.6: *********************** (4173 4%)
0.6-0.7: *************************** (3696 3%)
0.7-0.8: ***************************** (3378 3%)
0.8-0.9: ********************************* (3001 3%)
0.9-1.0: ************************************* (2691 2%)
```

Generador 2

```
0.0-0.1: *********** (8966 8%)
0.1-0.2: ************* (7601 7%)
0.2-0.3: *************** (6253 6%)
0.3-0.4: ****************** (5371 5%)
0.4-0.5: ********************* (4734 4%)
0.5-0.6: ********************** (4407 4%)
0.6-0.7: ************************** (3818 3%)
0.7-0.8: ********************************* (3004 3%)
0.8-0.9: ********************************* (2958 2%)
0.9-1.0: *************************************** (2510 2%)
```

Generador 1

Ejercicio 4

$$\theta = \int_{-\infty}^{\infty} e^{-x^2}\, dx = 1.7724538509055160 2$$

Tasks:

1. Transforme la integral a una con límites de 0 a 1, muestre su procedimiento

Tenemos la integral: $\int_{-\infty}^{\infty} e^{-x^2}\, dx$

Podemos separarlo: $\int_{-\infty}^{0} e^{-x^2}\, dx + \int_{0}^{\infty} e^{-x^2}\, dx$

Ahora calculamos: $\int_{-\infty}^{0} e^{-x^2}\, dx = \frac{\sqrt{\pi}}{2}$

$\int_{0}^{\infty} e^{-x^2}\, dx = \frac{\sqrt{\pi}}{2}$

como son iguales, utilizamos una expresión y multiplicamos (2)

Reescribimos: $2\int_{0}^{\infty} e^{-x^2}\, dx$

Sabemos que: $y = \frac{x-a}{b-a}$; $dy = \frac{dx}{b-a}$; $h(y) = \frac{g(\frac{1}{y}-1)}{y^2}$

Podemos reescribir como: $2\int_{0}^{\infty} e^{-x^2} dx = 2\int_{0}^{1} e^{-(\frac{1}{x}-1)^2} \cdot \frac{1}{x^2}\, dx$

∴ la integral con límites de 0 y 1 es: $2\int_{0}^{1} \frac{e^{-(\frac{1}{x}-1)^2}}{x^2}\, dx$

2. Aproxime la integral usando el método de Montecarlo, haciendo 100, 10,000 y 100,000 iteraciones.

```
N = 100 --> Monte Carlo Integration =  1.6346617811695856

N = 10,000 --> Monte Carlo Integration =  1.7707798099606746

N = 100,000 --> Monte Carlo Integration =  1.7698513643311664
```

Ejercicio 5

Tasks:

1. Transforme la integral múltiple a una en la que ambos límites sean de 0 a 1, muestre su procedimiento
2. Aproxime la integral usando el método de Montecarlo, haciendo 100, 10,000 y 100,000 iteraciones