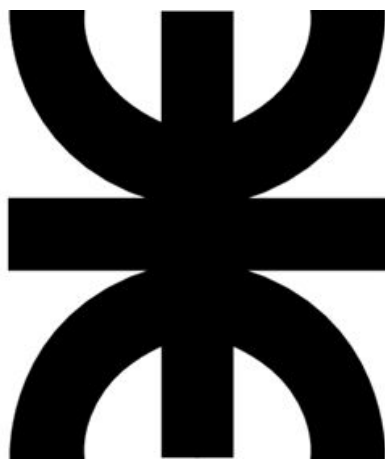


# **TRABAJO PRÁCTICO: FRACTALES**



**CANTEROS**, Lisandro Ariel

**GARAY**, Sebastián

**ZALEGAS**, Elías Leonel

**MATEMÁTICA SUPERIOR**

**INGENIERÍA EN SISTEMAS DE INFORMACIÓN**

**PROFESORES:**

Dr. Juan Eduardo **NÁPOLES VALDÉS**

Ing. Claudia Roxana **GARCÍA**

**UNIVERSIDAD TECNOLÓGICA NACIONAL**

**FACULTAD REGIONAL RESISTENCIA**

**NOVIEMBRE, 2020**

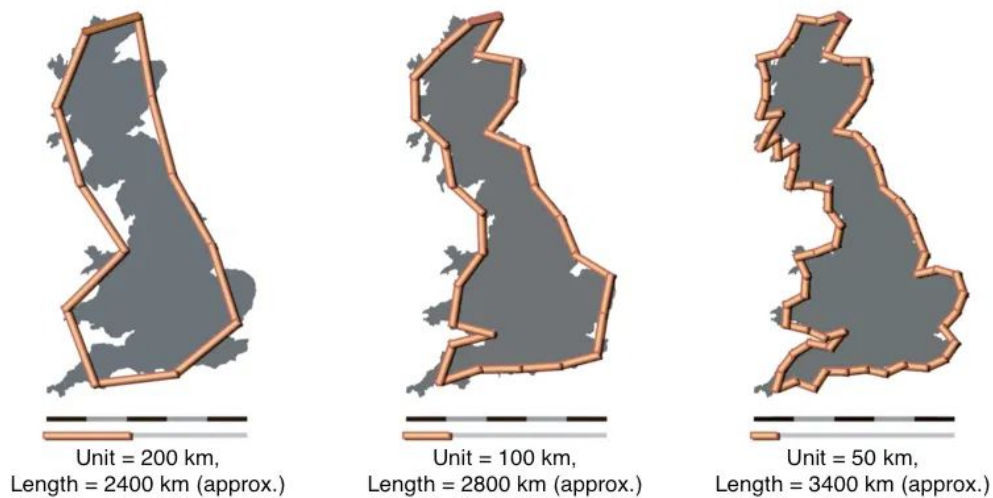
# CONTENIDO

Capítulos	Páginas
CONTENIDO	1
INTRODUCCIÓN	1
DESARROLLO	3
1. Fractales	3
2. Compresión Fractal de Imágenes	4
3. Generación por procedimiento	8
Hoja de trabajo 1	11
Hoja de trabajo 2	17
Hoja de trabajo 3	28
Hoja de trabajo 4	35
Hoja de trabajo 5	42
Hoja de trabajo 6	50
Hoja de trabajo 7	51
VIDEO	52
CONCLUSIONES	52
REFERENCIAS	52

## INTRODUCCIÓN

La naturaleza muestra gran variedad de formas las cuales poseen un alto nivel de complejidad que en muchos casos no es posible describir de forma precisa utilizando la geometría euclidiana. Incluso se pensó que algunos de estos fenómenos eran tan caóticos o aleatorios que describirlos matemáticamente era imposible.

Benoît Mandelbrot publicó, en 1967, su famoso artículo titulado “*How Long Is the Coast of Britain? Statistical Self-Similarity and Fractional Dimension*” (¿Qué tan larga es la costa de Gran Bretaña? Autosimilitud estadística y dimensión fractal), allí Mandelbrot plantea que la longitud de la línea costera depende de la escala con la que se la mida.



Los resultados demuestran que a medida que disminuye la escala, aumenta el perímetro, se podría seguir repitiendo este proceso hasta encontrar los átomos que componen la costa. Mandelbrot no decía que la costa era realmente fractal, él planteaba que podía comportarse como uno. Este tipo de contornos no puede ser representado con precisión mediante las conocidas figuras de la geometría euclidiana.

Como este ejemplo hay muchos más, Mandelbrot plantea en su libro *“The fractal geometry of nature”* que *“Las nubes no son esferas, las montañas no son conos, los litorales no son circulares, y los ladridos no son suaves, lo mismo que los relámpagos no viajan en línea recta.”*

La geometría fractal brinda la posibilidad de analizar fenómenos naturales que se pensaban como caóticos y aleatorios utilizando la matemática.



## DESARROLLO

### 1. Fractales

El término fractal deriva del latín *fractus* y significa quebrado o fracturado. Fue propuesto por Benoît Mandelbrot en 1975. Los fractales son objetos geométricos que repiten un

determinado patrón dando como resultado formas geométricas que se repiten de forma infinita, es decir son iterativos.

## **1.1 Características**

1. Autosimilitud/autosemejanza: según Mandelbrot, un objeto es autosimilar si todas sus partes tienen la misma estructura que el todo.
  - a. Autosimilitud exacta: el objeto geométrico es idéntico a cualquier escala, se dan en fractales producidos por funciones iteradas.
  - b. Autosimilitud aproximada: cuando luego de un cambio de escala, la forma de las partes presentan leves diferencias respecto de todo, generalmente se dan en la naturaleza.
  - c. Autosimilitud estadística: sólo se conservan algunas propiedades luego del cambio de escala.
2. Dimensión fractal/fraccionaria: dado que un fractal está compuesto por partes más pequeñas de sí mismo, la longitud, en teoría, sería infinita. Por lo tanto no se debe hablar de longitud de perímetro de un fractal, sino de su dimensión.

Un objeto es fractal cuando su dimensión fractal es mayor que su dimensión topológica

- a. Dimensión topológica: es aquella que dice si una figura se encuentra en la recta numérica (1 dimensión), en el plano (2 dimensiones) o en el espacio (3 dimensiones).
  - b. Dimensión fractal: representa el grado de irregularidad de un objeto geométrico, es una cantidad no entera.
3. Se generan a través de iteraciones sucesivas mediante un algoritmo que se ejecuta indefinidamente.
4. Los fractales no son diferenciables en ningún punto.

## **1.2 Tipos de fractales**

Lineales: son aquellos que pueden obtenerse mediante un proceso geométrico euclídeo. Algunos ejemplos son la curva de Koch, el triángulo de Sierpinski, el Conjunto de Cantor.

No lineales: son aquellos que se obtienen iterando infinitas veces una función de variable compleja, por ejemplo el Conjunto de Mandelbrot.

# **2. Compresión Fractal de Imágenes**

## **- 2.1 Un Poco de Historia**

En sí uno de los algoritmos de compresión fractal práctico más ampliamente conocido fue inventado por Alan Sloan y Michael Barnsley, en donde este último inició el desarrollo de la compresión fractal en 1987, como tal todos los métodos se basan en lo que se denomina “transformación fractal”, utilizando un sistema de funciones iteradas (que es lo mismo que se utiliza para la construcción de un triángulo de Sierpinski por ejemplo). Cabe agregar que Michael Barnsley y Alan Sloan crearon Iterated Systems Inc en 1987 donde crearon 20 patentes adicionales relacionadas con la compresión fractal.

Uno de los mayores pasos dados por la empresa fundada por ellos, Iterated Systems Inc. fue la creación de un proceso para hacer transformación fractal automática, el cual eliminó la necesidad de intervención humana durante la compresión, como era el caso que se daba anteriormente en la experimentación inicial con la tecnología de compresión fractal. La atención e interés generado por el uso de esta técnica fue tan popular que en 1992 Iterated Systems Inc. recibió una financiación de 2,1 millones de dólares del gobierno estadounidense para desarrollar un prototipo de chip para almacenamiento de imágenes digitales y descompresión utilizando tecnología de compresión de imagen mediante transformación fractal.

La compresión fractal de imagen se ha añadido en varias aplicaciones comerciales actuales como por ejemplo: “onDrive software”, desarrollado bajo licencia de Iterated Systems Inc, “Genuine Fractals 5,” que es un plugin de Photoshop capaz de grabar archivos en formato comprimido FIF (Fractal Image Format). Uno de los usos más exitosos de la compresión fractal de imágenes fue de parte de Microsoft, en su antigua enciclopedia multimedia “Encarta”, aunque ya este ha sido descontinuado.

## - 2.2 Definición, Propósito, Funcionamiento

En la vida real a la hora de querer guardar imágenes en algún almacenamiento, existe un gran problema, y es el del tamaño que ocupan estas mismas, para poder solucionar esto lo que se hace es recurrir a “técnicas de Compresión”, es decir cualquier procedimiento de codificación que tenga como objetivo representar cierta cantidad de información utilizando una menor cantidad de la misma, siendo imposible una reconstrucción exacta de los datos originales.

Existen dos tipos de compresión, *con pérdida* y *sin pérdida* (en lugar de guardar una copia exacta, solo se guarda una aproximación. Esta aproximación se aprovecha de las limitaciones de la percepción humana para esconder la distorsión introducida.). La compresión fractal utiliza este último nombrado y tal como dice su nombre, está basado en fractales. La compresión fractal está orientada más que nada en lo que es texturas e imágenes naturales, centrándose en el hecho de que partes de una imagen, a menudo, se parecen a otras partes de la misma imagen. Los algoritmos fractales convierten estas partes en datos matemáticos llamados *códigos fractales* los cuales se usan para recrear la imagen codificada.

Absolutamente toda la compresión fractal está basada específicamente en lo que se llama *sistema particionado de funciones iteradas* (PIFS por sus siglas en inglés) la cual brinda una compresión de imagen sorprendentemente buena, incluso para fotografías que no parecen

tener el tipo de estructura auto-similar que muestran los fractales IFS (sistema iterativo de funciones), para llevar a cabo el PIFS, se utiliza el algoritmo: *quadtree partitioning*

Supongamos que queremos comprimir la siguiente imagen:



Hacemos los primeros pasos del algoritmo. Partimos la imagen en cuatro celdas rango  $R_i$ 's y al mismo tiempo hacemos una base de celdas dominio  $D_i$ . La base de las  $D_i$ 's debe ser muy grande, por lo que hacemos que las celdas se sobrelapen. Dicha base debe contener celdas de todos los tamaños, para poder estar seguros de que siempre encontremos una  $D_i$  que cubra a una celda rango dada.



$R_i$ 's

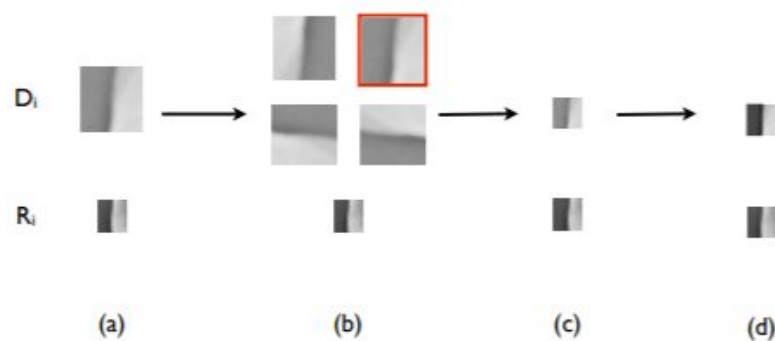


$D_i$ 's

Vemos un candidato para cubrir la celda rango. Tarde o temprano en el algoritmo de partición quadtree se llega a una celda rango como esta  $R_i$ . El algoritmo busca en la base de celdas dominio y encuentra que esta  $D_i$  la cubre bien.



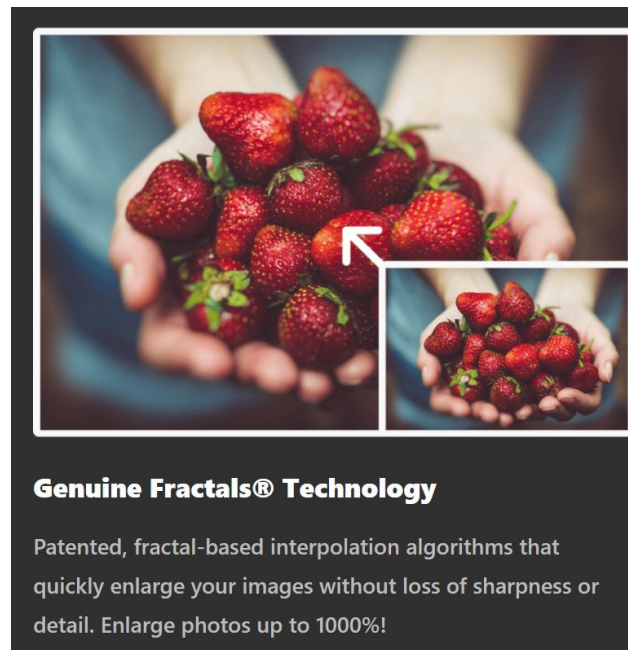
Primero: Tenemos la celda rango y el dominio que queremos comparar. Segundo: rotamos a  $D_i$  en una de las ocho posibles formas (cada  $90^\circ$  es una, también reflejamos y rotamos otras cuatro veces) y escogemos la mejor. , aquí solamente mostramos cuatro posibles rotaciones. Tercero: comprimimos a  $D_i$  para que coincida con el tamaño de  $R_i$ . Cuarto: ajustamos el contraste y el brillo por medio de mínimos cuadrados para obtener sus valores óptimos.



## - 2.3 Ejemplos Prácticos y aplicaciones Reales

Una de las diferencias fundamentales de este tipo de técnicas es que el código de compresión no guarda píxeles, por lo que es libre de escalas , de esta forma se puede descomprimir a

cualquier escala sin tener problemas de resolución. En otras palabras, la independencia de la resolución de una imagen codificada en fractal puede utilizarse para incrementar la resolución de una imagen. Este proceso se conoce como "interpolación fractal". En la interpolación fractal, una imagen se codifica a código fractal mediante compresión fractal, y posteriormente descomprimida a una resolución superior.



*Imagen promocional del uso de "interpolación fractal" en el software "Perfect Resize 7"*

### 3. Generación por procedimiento

La generación por procedimiento o generación procedural es un método de creación de contenidos a través de algoritmos, en oposición a un método de creación manual, y se lo aplica tanto en simulaciones gráficas por computadora como en videojuegos, instalaciones, programación y en música.

Los fractales pueden ser un ejemplo de animación por generación procedural: funciones matemáticas gráficas repetidas hasta el infinito. El sonido digital también se puede generar de forma procedural, un uso que se ha desarrollado mucho para la música electrónica: ejemplos de esto se encuentran en la obra de Brian Eno con la popularización del término "música generativa".

Enfocándonos principalmente en los videojuegos, la generación por procedimientos, como mencionamos anteriormente, se basa en la creación automática y aleatoria del contenido de un juego gracias al uso de algoritmos; tales contenidos pueden ir desde el núcleo de la jugabilidad, los niveles y desafíos, hasta los aspectos más ilimitados como las armas, texturas, personajes, ítems, escenarios, etc.



### - 3.1 Un Poco de Historia

Antes de los videojuegos orientados gráficamente, podemos comenzar por juegos de 1978 de *Beneath Apple Manor* que utilizaron la generación por procedimientos para la construcción de mazmorras para los sistemas ASCII (aunque Beneath Apple Manor utilizaron gráficos Lo-Res de 16 colores). La generación por procedimientos podía definir habitaciones, pasillos, monstruos y tesoros que el jugador podría encontrar. Juegos basados en los conceptos roguelike siguen siendo un género popular en la actualidad, ya que permite el desarrollo de un juego complejo sin tener que gastar demasiado tiempo en la creación de un mundo de juego.

Los primeros videojuegos gráficos fueron severamente frenados por limitaciones de memoria. Esto provocó que el contenido, como mapas, fuera generado algorítmicamente sobre la marcha: porque simplemente no había espacio suficiente para almacenar una gran cantidad de niveles preconfeccionados y arte de los niveles. Generadores con números pseudoaleatorios fueron utilizados a menudo con valores predefinidos de semillas aleatorias con el fin de crear grandes mundos de juego que parecieran prefabricados.

Tal vez el primer juego en utilizar una semilla para generar el mundo es *Akalabeth* (1980) de Richard Garriott; en el cual utilizando el mismo número de semillas, un jugador siempre puede volver a un mundo dado. *The Sentinel* supuestamente tiene 10.000 niveles diferentes almacenados en solo 48 y 64 kilobytes. Un caso extremo fue *Élite*, que fue planeado originalmente para contener un total de 248 (aproximadamente 282.000.000.000.000) galaxias con 256 sistemas solares cada uno. El editor, sin embargo, tenía miedo de que un universo tan gigantesco causaría incredulidad en los jugadores, y solo ocho de estas galaxias se eligieron para la versión final. Otros de los primeros ejemplos notables incluyen el juego de 1985 *Rescue on Fractalus* que utiliza fractales para crear con procedimientos en tiempo real las montañas escarpadas de un planeta alienígena y *River Raid*, el juego de 1982 de *Activision* que utiliza una secuencia de números pseudoaleatorios generados por un registro de desplazamiento lineal de realimentación con el fin de generar un laberinto desplazable de obstáculos.

A medida que avanzaba el hardware y los CD fueron capaces de almacenar miles de veces más datos de lo que era posible en los años 80, usar generación por procedimientos para construir mundos grandes se hicieron innecesarios. El contenido del juego como las texturas, modelos de los personajes y el medio ambiente son a menudo creados por artistas de antemano, para mantener la calidad del contenido consistentemente alta.

Almacenando diseños únicos en la memoria del juego, en lugar de producir múltiples objetos usando directrices similares lo que permite una variedad de modelos detallados siendo cada uno sustancialmente diferente. Sin embargo, cuando cada modelo tiene que ser diseñado a mano, existirá un número limitado de objetos. Algunas aproximaciones iniciales a la síntesis de procedimientos intentaron resolver este problema mediante la inversión de la carga de la generación de contenidos de los artistas a los programadores los cuales pueden

crear código que genera automáticamente diferentes mallas de acuerdo a los parámetros de entrada.

Aunque a veces esto todavía sucede, lo que se ha reconocido es que la aplicación de un modelo puramente procedimental es a menudo difícil ya que requiere enormes cantidades de tiempo para convertirse en un método funcional, usable y de aspecto realista. En vez de escribir un procedimiento que solo construye contenido procedimentalmente, se ha demostrado que es mucho más barato y eficaz que delegar al artista la creación de contenido para algunos detalles. Por ejemplo, *SpeedTree* se utiliza para generar una gran variedad de árboles generación por procedimientos, sin embargo, sus texturas de las hojas se pueden extraer de archivos regulares, a menudo representando digitalmente follaje real. Otros métodos eficaces para generar contenidos híbridos son los de fusionar procedimentalmente diferentes activos preconfeccionados o aplicar con esta técnica algunas distorsiones a ellos.

Suponiendo, sin embargo, un único algoritmo se puede prever para generar un árbol de aspecto realista, el algoritmo podría ser llamado para generar árboles al azar, llenando así todo un bosque en tiempo de ejecución, en lugar de almacenar todos los vértices requeridos por los diversos modelos. Esto ahorraría espacio en los medios de almacenamiento y reduciría la carga de los artistas, al tiempo que proporcionaría un nivel similar de inmersión al jugador. Este método requeriría mucha más potencia de procesamiento, pero como los CPUs están en constante aumento de velocidad no es un obstáculo.

Un problema diferente es que no es fácil desarrollar un buen algoritmo para un solo árbol, y mucho menos para una variedad de especies (comparar sumac, abedul, arce). Una advertencia adicional es que el montaje de un bosque de aspecto realista no podía hacerse simplemente montando árboles porque en el mundo real hay interacciones entre los diferentes árboles que pueden cambiar dramáticamente su apariencia y distribución.

## - 3.2 Actualidad

A pesar de todos los acercamientos que tuvo la generación por procedimiento en la historia, el juego que lograría dar el impulso definitivo a la generación por procedimiento y llevarla, de nuevo, al primer plano del desarrollo de videojuegos fue *Minecraft* (2009), en este juego los millones de cubos son automáticamente generados por los algoritmos.

Otros juegos mucho más ambiciosos están utilizando la generación por procedimiento para crear universos completos con un elevado grado de complejidad. Por ejemplo, *No Man's Sky* tiene la posibilidad de generar 18 trillones de planetas para que el jugador se divierta explorando. La generación por procedimientos le permite a los creadores de *No Man's Sky* ofrecer un número casi infinito de planetas a un costo de producción muy bajo. La tecnología está siendo aprovechada sobre todo por las pequeñas compañías independientes para crear juegos como *Terraria* o *Salt*.

Lo más interesante de toda esta historia es que la generación por procedimientos no sólo sirve para construir mundos virtuales: en *Rimworld* la generación por procedimientos se

está usando para crear eventos y mini-narrativas. O sea, Rimworld funciona a partir de una “inteligencia artificial narrativa ” (AI Storyteller). Moon Hunters, por su parte, usa la PG para generar mitos.

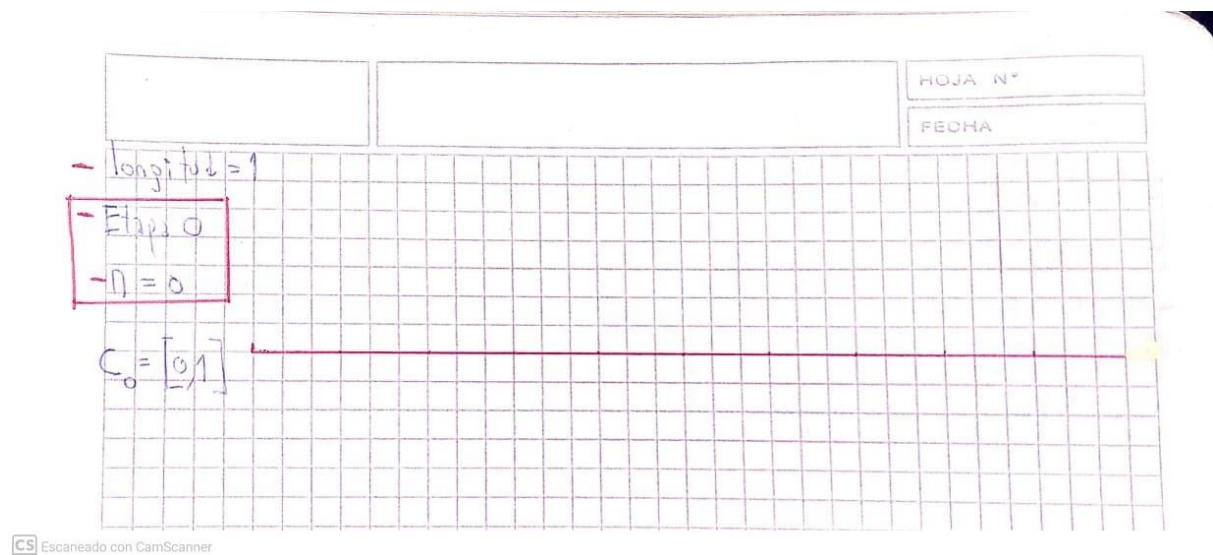
En los mundos virtuales creados con PG la vida sigue más allá de que haya alguien frente a la pantalla. Los universos artificiales evolucionan y sus actores continúan desarrollando su vida.

# Hoja de trabajo 1

## Ejercicio 1

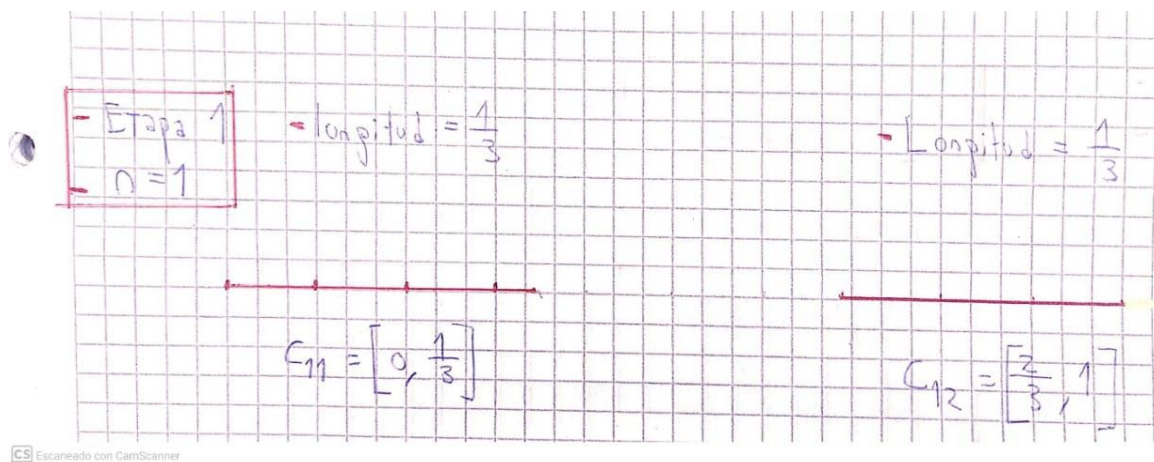
- A. Etapa 0,  $n = 0$ . Dibuje un segmento de longitud unitaria ( $l=1$ ) y llámelo

$C_0 = [0, 1]$ .



- B. Etapa 1,  $n=1$ . Divida el segmento en tres subsegmentos de longitud  $1/3$  cada uno, borre el segmento central, excepto los extremos y quédese con los segmentos restantes, Llame a esto  $C_{11}=[0, 1/3]$ ;  $C_{12}=[2/3, 1]$

B1. ¿Cual es la longitud de cada segmento remanente?



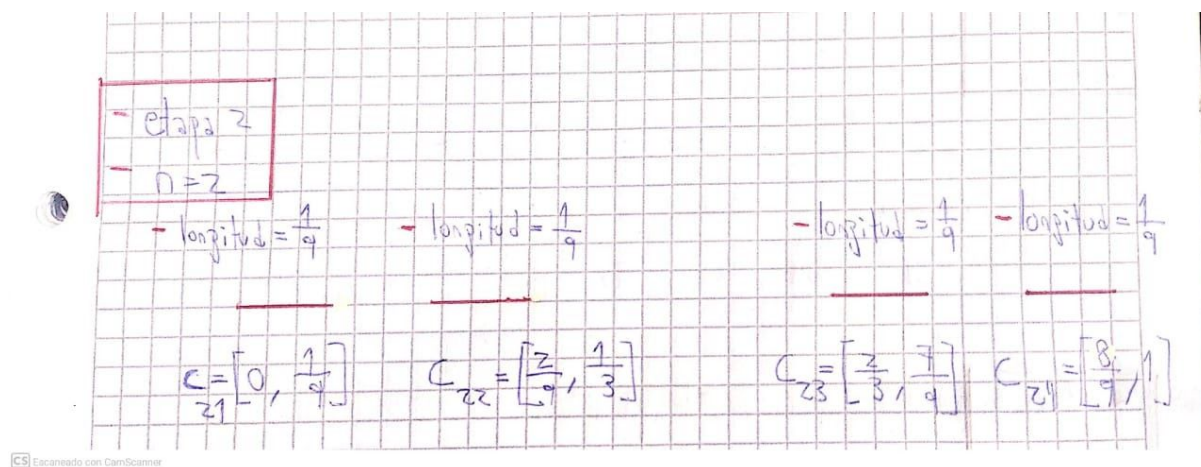
- C. Etapa 2, repita, nuevamente esta operación: a cada segmento remanente dividirlo en tres subsegmentos y deseche el tercio medio sin incluir los extremos

C1. ¿Cual es la longitud de cada segmento remanente?

**C2. Complete los siguientes cuatro subintervalos generados**

C21=[----,----] C22=[----,----] C23=[----,----] C24=[----,----]

**C3. ¿Cual es el valor de n? ¿Cuánto vale la longitud l de cada subintervalo?**



**D. Repita el algoritmo para n=3, n=4 y complete la tabla 1:**

Etapas	Nº de intervalos remanentes	Longitud de cada intervalo
0	1	1
1	2	1/3
2	4	1/9
3	8	1/27
4	16	1/81

**E. ¿Cuántos subintervalos hay en la etapa 5? y ¿en la etapa 6? ¿Cual es la longitud de cada subsegmento en dichas etapas?**

En la etapa 5 existen 32 subintervalos (simplemente hacemos  $2^5$ ) con una longitud de  $\frac{1}{243}$  (lo que hacemos es dividir por 3 la longitud de la etapa anterior, es decir  $\frac{1}{3} \cdot \frac{1}{81}$ ). Proseguimos de la misma manera con la etapa 6, quedándonos como que tiene 64 intervalos, con una longitud de  $\frac{1}{729}$  cada uno

- **F. Generalice para encontrar el número de intervalos y la longitud para la etapa  $n$**

La forma de encontrar una fórmula genérica para hallar tanto los números de intervalos y longitud en una etapa  $n$  es muy parecida a la forma de hallar cada solución que utilizamos en el punto anterior. Para hallar el número de intervalos se utiliza la fórmula:  $2^n$  y para hallar la longitud del intervalo se utiliza la fórmula  $(\frac{1}{3})^n$

- **G. Cuando  $n$  crece indefinidamente, ¿que sucede con el número de intervalos y la longitud de los mismos?**

Cuando un “ $n$ ” crece indefinidamente lo que sucede es que el número de intervalos se hace extremadamente alto, creciendo también indefinidamente, es decir tiende a  $\infty$ . En cuanto las longitudes de los intervalos, se hacen cada vez más pequeñas, es decir tienden a 0

## Ejercicio 2

### Enuncie el algoritmo para construir el fractal de Cantor

El algoritmo que se tiene que llevar a cabo para la realización del fractal de Cantor (para un conjunto de Cantor de  $1/3$ ) es tal como se explicaba en la primera parte de la hoja de trabajo, el siguiente:

Primero lo que se hace es que a partir del intervalo  $I_0 = [0, 1]$  (que es un intervalo cerrado de longitud 1) se lo parte en 3 partes exactamente iguales, es decir subdividirlo en partes de longitud  $\frac{1}{3}$  cada una, luego se eliminaría el subintervalo que se encuentra en el medio de los otros 2, quedando como resultado lo que se denomina “la primera iteración” del algoritmo, que es  $I_1 := [0, 1/3] \cup [2/3, 1]$  (en otras palabras es la unión de los 2 intervalos que quedaron).

Para proseguir con el algoritmo, se procede de la misma manera con exactamente la misma subdivisión de en 3 partes partes iguales, pero ahora con los subintervalos finales de la “primera iteración” obteniendo de esta forma nuevos subintervalos que serían  $I_2 = [0, 1/9] \cup [2/9, 1/3] \cup [2/3, 7/9] \cup [8/9, 1]$  los cuales tienen una longitud de  $1/9$  cada uno, este nuevo conjunto de intervalos que obtenemos se lo denomina “la segunda iteración” del algoritmo.

Se procede de igual manera infinitas veces, para obtener infinitas iteraciones como tal, este sería el algoritmo para la construcción del fractal de Cantor

### **Ejercicio 3**

- **A. Usted puede construir otro conjunto de Cantor dividiendo un segmento, al que se le asigna longitud 1, en cinco subsegmentos. Elimine el intervalo  $\frac{2}{5} < x < \frac{3}{5}$  del intervalo  $0 < x < 1$  en la primera etapa. Luego remueva los quintos medios de cada intervalo remanente, y así sucesivamente. Realice un listado de los intervalos que fueron removidos hasta la etapa 3**



**etapa 0**  
 $n=0$   
 longitud = 1  
 $C_0 = [0, 1]$

---

**etapa 1**  
 $n=1$   
 longitud =  $\frac{2}{5}$   
 $C_{11} = [0, \frac{2}{5}]$        $C_{12} = [\frac{3}{5}, 1]$

---

**etapa 2**  
 $n=2$   
 longitud =  $\frac{4}{25}$       longitud =  $\frac{4}{25}$       longitud =  $\frac{4}{25}$       longitud =  $\frac{4}{25}$   
 $C_{21} = [0, \frac{4}{25}]$        $C_{22} = [\frac{6}{25}, \frac{2}{5}]$        $C_{23} = [\frac{3}{5}, \frac{14}{25}]$        $C_{24} = [\frac{24}{25}, 1]$

---

**etapa 3**  
 $n=3$   
 longitud todos los intervalos =  $\frac{8}{125}$

(a)	(b)	(c)	(d)	(e)	(f)	(g)	(h)
(a) $C_{31} = [0, \frac{8}{125}]$	(b) $C_{32} = [\frac{12}{125}, \frac{4}{25}]$	(c) $C_{33} = [\frac{5}{25}, \frac{38}{125}]$	(d) $C_{34} = [\frac{12}{125}, \frac{2}{5}]$	(e) $C_{35} = [\frac{3}{5}, \frac{83}{125}]$	(f) $C_{36} = [\frac{87}{125}, \frac{14}{25}]$	(g) $C_{37} = [\frac{24}{25}, \frac{113}{125}]$	(h) $C_{38} = [\frac{117}{125}, 1]$

Los intervalos que fueron removidos son: en la etapa 1 solo un intervalo, en la etapa 2 dos intervalos, en la etapa 3 cuatro intervalos.

- **B. Complete la Tabla 2.**



Etapas	Nro de intervalos remanentes	longitud del intervalo desechado	longitud de los intervalos remanentes
1	2	1/5	2/5
2	4	2/25	4/25
3	8	4/125	8/125
4	16	8/625	16/625

- **C. ¿Cuántos subintervalos hay en la etapa 3? y ¿en la etapa 4? ¿Cual es la longitud de cada subsegmento en dichas etapas?**

En la etapa 3 existen 8 subintervalos remanentes con una longitud de  $\frac{8}{125}$  cada uno y en la etapa 4 existen 16 subintervalos remanentes con una longitud de  $\frac{16}{625}$  cada uno

- **D. Generalice para encontrar el número de intervalos remanentes y su longitud para la etapa n**

En sí a la hora de generalizar una forma para hallar el número de intervalos remanentes en una etapa n (pero en un conjunto de cantor de 1/5) es exactamente la misma fórmula para hallar el número de intervalos remanentes en un conjunto de cantor de 1/3, es decir " $2^n$ ", ahora para saber una fórmula generalizada de la longitud de estos intervalos en una etapa n, tampoco se diferencia mucho de la forma que utilizabamos en el conjunto de cantor de cantor de 1/3 de nuevo, siendo esta  $\left(\frac{2}{5}\right)^n$

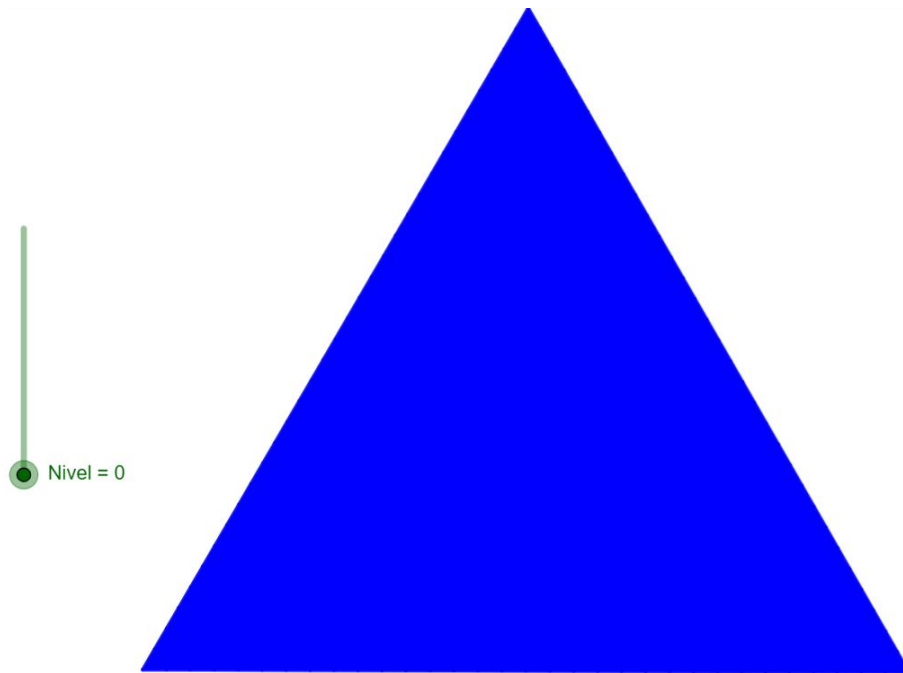
- **E. Cuando n crece indefinidamente , ¿que sucede con el número de intervalos y la longitud de los mismos?**

Nuevamente aquí sucede lo mismo que con lo que pasaba con el conjunto de cantor de 1/3, el número de intervalos sigue creciendo indefinidamente, es decir tiende a  $\infty$  y en cuanto las longitudes de los intervalos, se hacen cada vez más pequeñas, es decir tienden a 0

## Hoja de trabajo 2

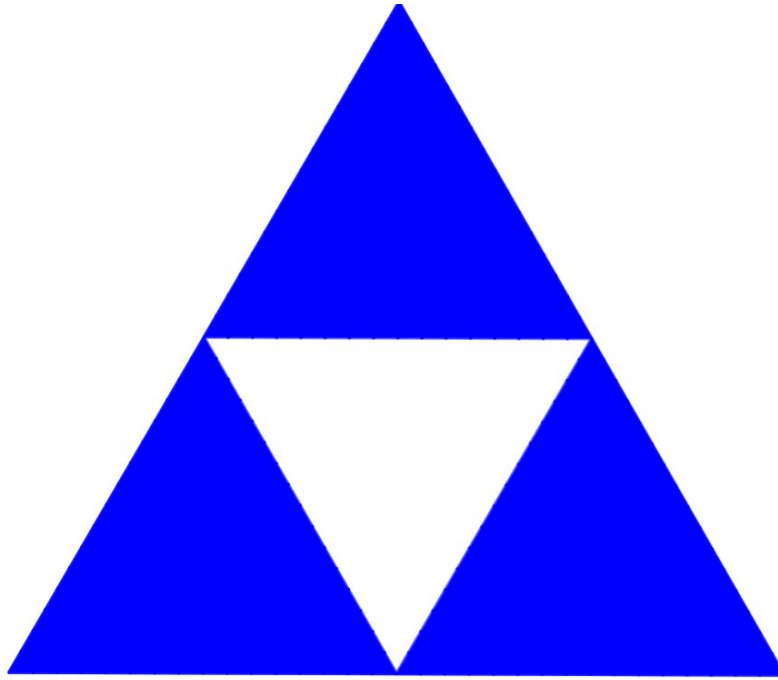
### Ejercicio 1

Originalmente se tiene un triángulo equilátero como tal



Para comenzar con el algoritmo cuyo objetivo es la de obtener el triángulo de Sierpinski, se parte el triángulo original en 4 triángulos los cuales estarían dentro del triángulo inicial como tal, para que luego se elimine el triángulo central (que es el único triángulo invertido por así decirlo) en otras palabras los triángulos remanentes son 3, obteniendo la siguiente figura:

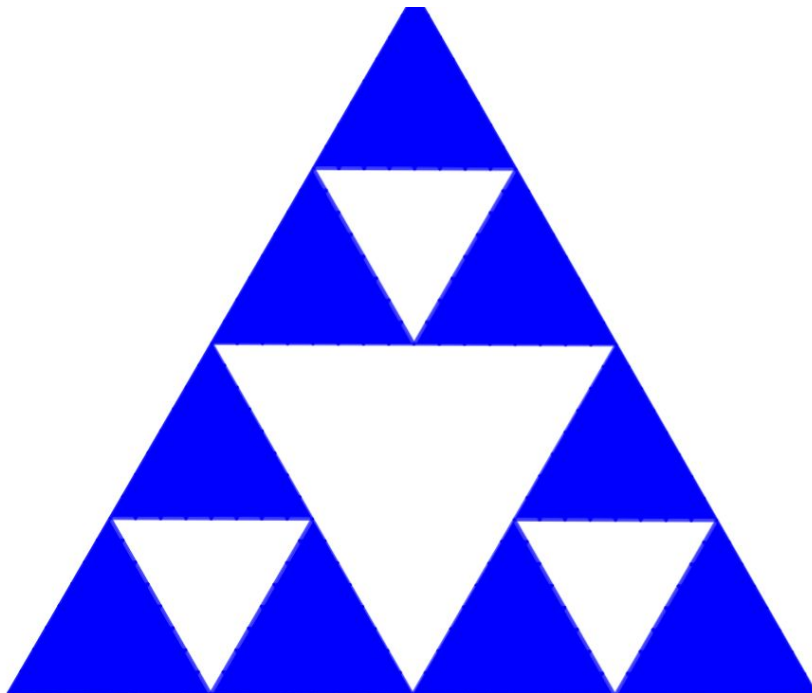
Nivel = 1



- Repita en cada subtriángulo el proceso y deseche nuevamente el triángulo central inferior. ¿cuántos triángulos remanentes quedan?

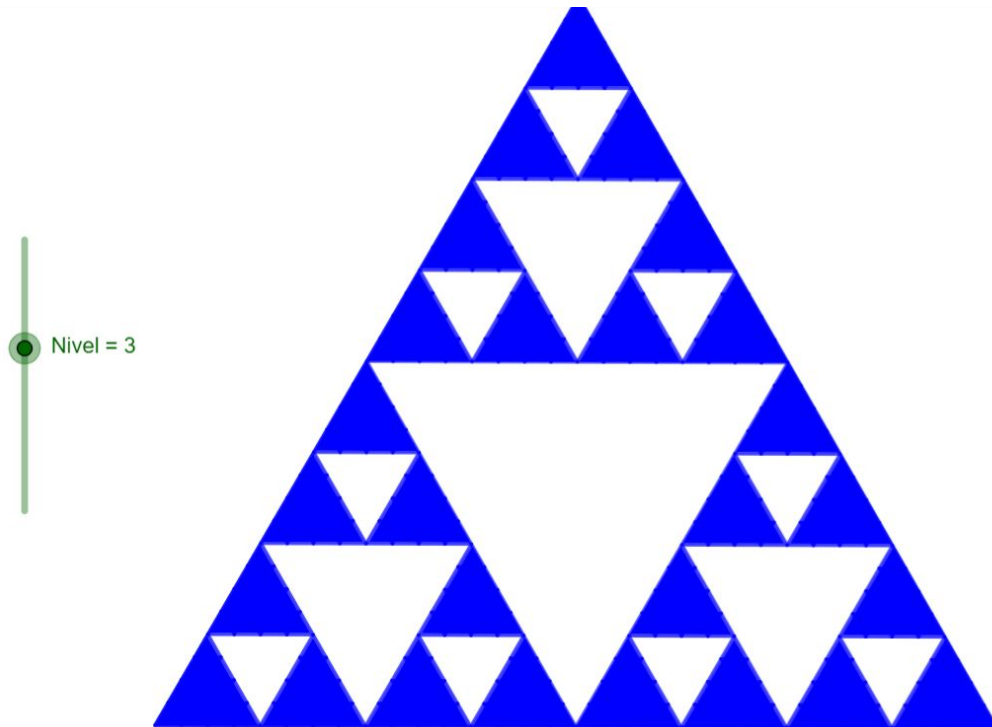
Procediendo nuevamente de la misma manera con los triángulos remanentes anteriores, vemos que ahora nos van a quedar 9 triángulos remanentes.

Nivel = 2

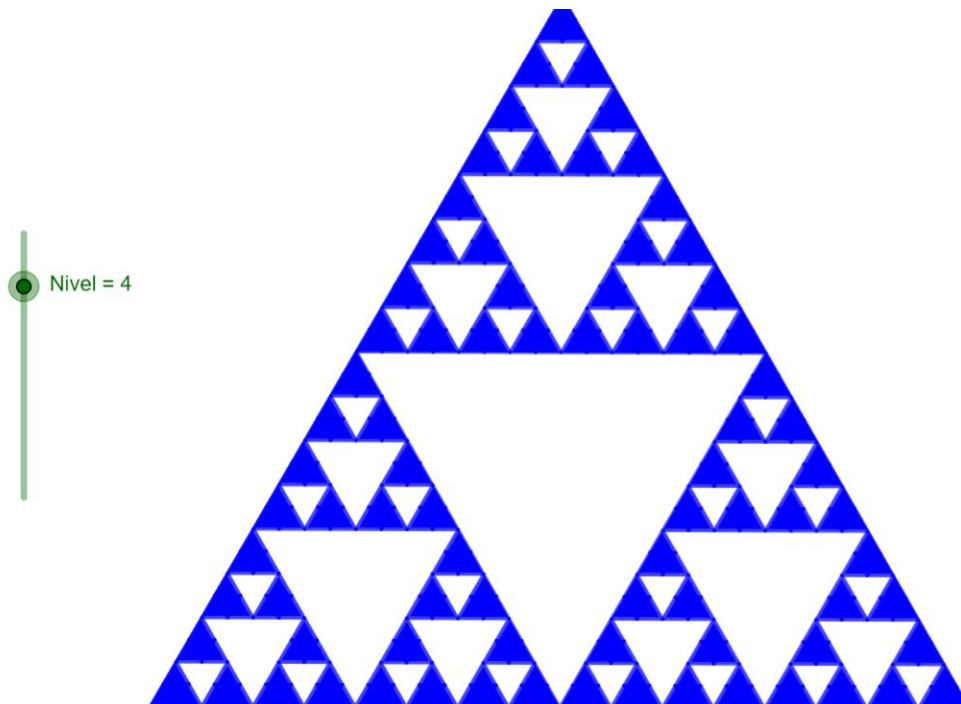


- Repita el algoritmo dos veces más. ¿Cuántos triángulos remanentes obtuvo en cada etapa?

Tal como nos dice la consigna, haremos 2 veces más el procedimiento, en esta ocasión quedándonos 27 triángulos remanentes:



luego 81 triángulos remanentes:



- **A. ¿Qué ocurriría si se repite el proceso una y otra vez? ¿Cómo cambia la figura? Si el proceso continuará indefinidamente obtendremos el triángulo de Sierpinski**

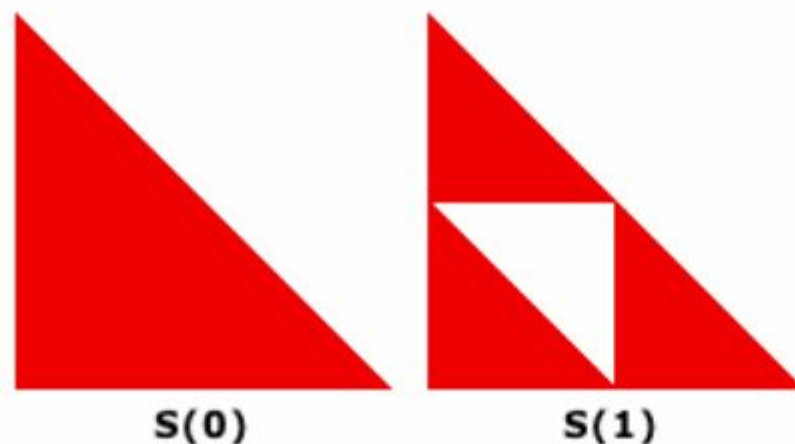
Si se repite el proceso una y otra vez, cada uno de los triángulos remanentes que se tienen actuales, se convierten en 3 más. prosiguiendo de la misma manera, las veces que se quiera y haciendo al triángulo original que se tenía al principio, un conjunto de triángulos cada vez más pequeños.

- **B. Explique qué pasaría si en lugar de conservar los tres triángulos de las esquinas, se aplica el algoritmo para el triángulo central inferior hasta la etapa 5**

En el caso de que se aplicara el mismo algoritmo pero con el triángulo del centro. es decir, conservando el triángulo central y borrando los triángulos de las esquinas, lo que podríamos observar es un solo triángulo el cual con el paso de las diferentes etapas, se va volviendo más pequeño, y en donde se invierte con cada una de estas etapas como tal.

- **C. Repita el algoritmo desde la etapa 1 a la 3 para un triángulo rectángulo e isósceles, e informe lo que observa**

### Triángulo Rectángulo



Tal como se puede ver, es totalmente posible hacer un triángulo de Sierpinski con triángulos rectángulos, al realizar la etapa 1 del algoritmo, el triángulo del centro que se lo saca, es también un triángulo rectángulo pero invertido



**S(2)**

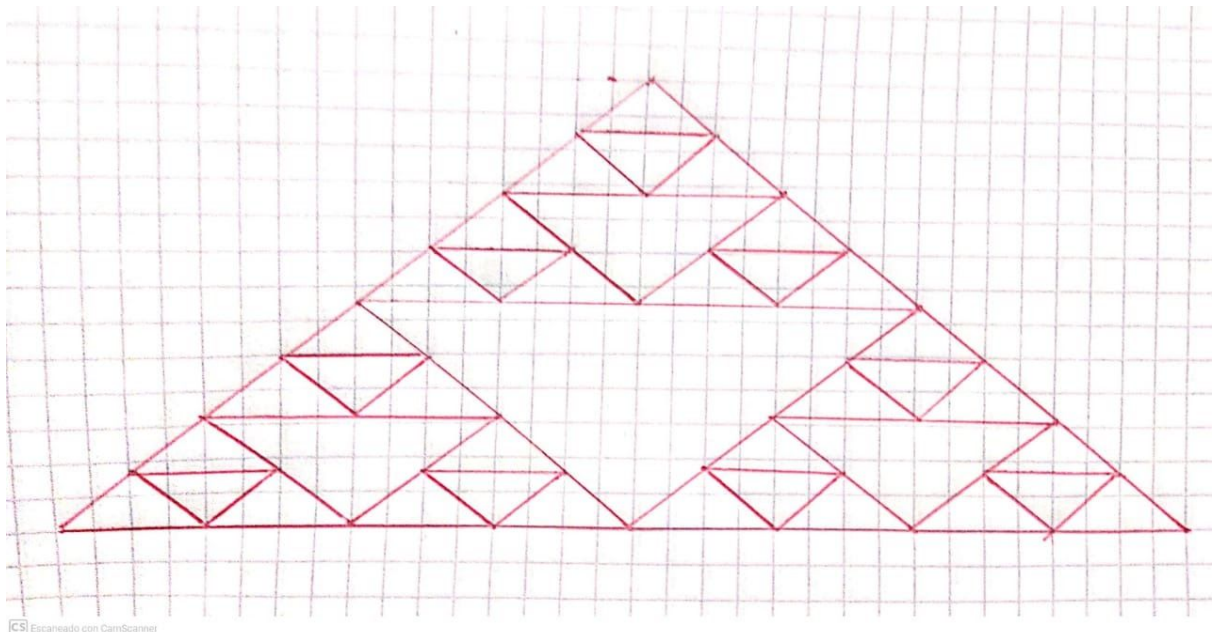


**S(3)**

Como se observa en las 2 siguientes etapas, puede notarse que progresivamente la figura se va pareciendo a cuando usábamos triángulos equiláteros pero como si esta estuviera ligeramente inclinada hacia la izquierda (porque usamos triángulos rectángulos cuya hipotenusa apunta a la derecha, si fuese al revés, veríamos que en sí el triángulo de Sierpinski se inclina hacia la derecha).

### **Triángulo Isósceles**

En cuanto este tipo de triángulo, es posible también aplicar el algoritmo para hacer un triángulo de Sierpinski, en donde el triángulo central el cual se quita es un triángulo isósceles invertido.



En cuanto a la forma que va teniendo el triángulo de Sierpinski progresivamente, se ve también como si se estuviera utilizando un triángulo equilátero, solamente que más achatada o aplastada (en donde se va a ver más aplastada o no dependiendo de los ángulos de la base).

## Ejercicio 2

**Una modificación en el algoritmo permite construir un fractal distinto a los dos anteriores: divida cada lado del triángulo equilátero en tres partes iguales, guarde los seis subtriángulos que tienen un lado en alguno de los bordes.**

- **A. Repita el proceso realizando una segunda iteración en cada uno de los 6 subtriángulos remanentes. ¿Cuántos subtriángulos quedan ?**

Si repetimos el mismo proceso con una segunda iteración, nos quedaría la siguiente figura, en donde los triángulos remanentes son 36



- **B. Repita el proceso y recuerde que en cada etapa, cada triángulo es transformado en seis nuevos subtriángulos cuyo lado es un tercio de la longitud anterior, si la iteración continúa indefinidamente ¿que se observaría ?**

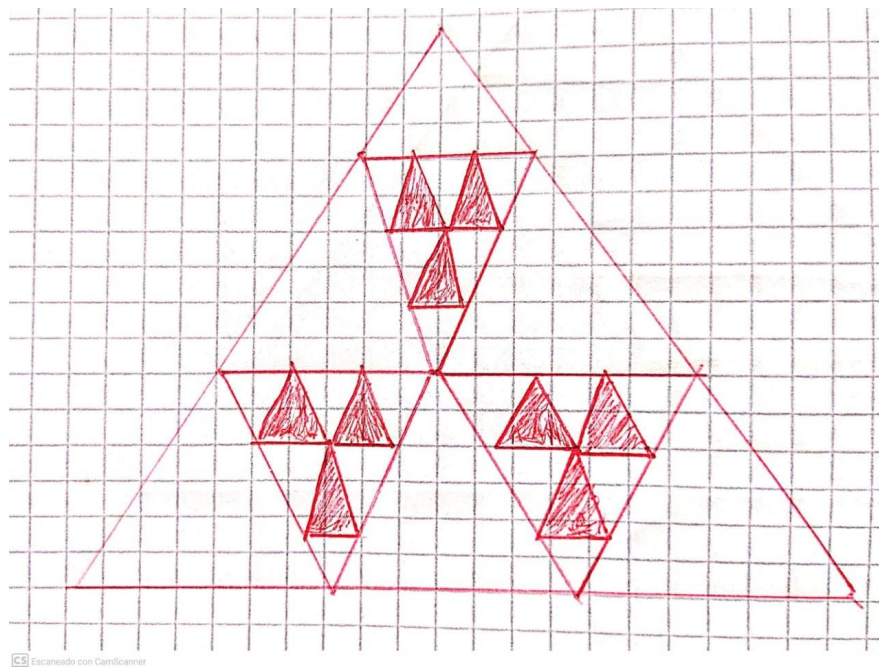
Si volvemos a repetir el proceso con otra iteración más, lo que nos queda es la siguiente figura (con 216 triángulos remanentes)



si se repite exactamente lo mismo indefinidamente, podemos observar que cada vez, progresivamente los triángulos cada vez son más y más, es decir tienden a infinito, al igual que con lo que sucede con el triángulo de Sierpińsk que no esta modificado

- **C. Explique qué sucedería si en lugar de guardarlos seis subtriángulos del borde, guardara los tres interiores, ¿que figura aparece después de dos iteraciones?**

Lo que sucede en el caso de que el algoritmo se base en que borremos los triángulos exteriores y dejemos los triángulos interiores, estos últimos se van a ir invirtiendo en cada una de las etapas y se volverán tanto más pequeños como más numerosos (en cada uno de los triángulos, en cada etapa, se subdividen en 3 más) progresivamente.





### Ejercicio 3

- **B. Complete la tabla según las etapas del triángulo de sierpinski:**

Etapas	Número de triángulos sombreados	Área de cada triángulo	Área total sombreada
0	1	1	1
1	3	1/4	3/4
2	9	1/16	9/16
3	27	1/64	27/64
4	81	1/256	81/256
5	243	1/1024	243/1024

- **C. ¿Cuál es la constante multiplicativa utilizada para encontrar el número de triángulos de una etapa a otra? ¿y en qué caso del área ?**

La constante utilizada para encontrar el número de triángulos remanentes en cada etapa es el “3”, se va multiplicando por este número, una y otra vez en cada etapa. En el caso del área de un triángulo es el numero “1/4” el cual se va multiplicando también en cada etapa

- **D. Encuentre las expresiones para calcular el número de triángulos. el del área de cada uno de ellos en la etapa n y el área total sombreada**

La expresión para hallar el numero de triangulos remanentes en una etapa n es :  $3^n$

La expresión para hallar la área de cada triángulo en una etapa n es:  $\left(\frac{1}{4}\right)^n$

La expresión para hallar el total de área de todos los triángulos en una etapa n es:  $\left(\frac{3}{4}\right)^n$

- **E. Cuando n aumenta indefinidamente, ¿Qué sucede con el número de triángulos? ¿que ocurre con el área de cada triángulo? ¿con el área total sombreada? ¿y con el área sin sombrear?**

Cuando n crece indefinidamente, lo que sucede con los números de triángulos que se van creando es que van a crecer indefinidamente también, es decir van tendiendo a infinito, en

cuanto las áreas de cada triángulo, van disminuyendo progresivamente en cuanto mayor sea la etapa  $n$ , trayendo como consecuencia que la sumatoria total de todas las áreas de cada triángulo también se vaya disminuyendo. Si nos referimos a los triángulos sin sombrear (los que se quitan) una vez que se los quita estos no cambian, es decir son estáticos y a medida que pasan las  $n$  etapas las áreas totales de estos mismos van creciendo.

## Ejercicio 4

Realice la misma actividad que en el ejercicio 3, pero con un triángulo rectángulo

- **B. Complete la tabla según las etapas del triángulo de sierpinski:**

Etapas	Número de triángulos sombreados	Área de cada triángulo	Área total sombreada
0	1	1	1
1	3	$1/8$	$3/8$
2	9	$1/64$	$9/64$
3	27	$1/512$	$27/512$
4	81	$1/4096$	$81/4096$
5	243	$1/32768$	$243/32768$

- **C. ¿Cuál es la constante multiplicativa utilizada para encontrar el número de triángulos de una etapa a otra? ¿y en qué caso del área ?**

La constante utilizada para encontrar el número de triángulos remanentes en cada etapa es la misma que es utilizada en el triángulo de sierpinski original, es decir “3”. En cuanto el área de cada triángulo, la constante es “ $1/8$ ” (porque en la etapa 1, el área de cada triángulo es  $\frac{1/2 * 1/2}{2}$ ) la cual se va multiplicando una y otra vez en cada etapa.

- **D. Encuentre las expresiones para calcular el número de triángulos. el del área de cada uno de ellos en la etapa  $n$  y el área total sombreada**

La expresión para hallar el número de triángulos remanentes en una etapa  $n$  es :  $3^n$

La expresión para hallar la área de cada triángulo en una etapa  $n$  es:  $\left(\frac{1}{8}\right)^n$

La expresión para hallar el total de área de todos los triángulos en una etapa  $n$  es:  $\left(\frac{3}{8}\right)^n$

- **E. Cuando  $n$  aumenta indefinidamente, ¿Qué sucede con el número de triángulos? ¿que ocurre con el área de cada triángulo? ¿con el área total sombreada? ¿y con el área sin sombrear?**

En esta variación, sucede exactamente lo mismo que con el triángulo de sierpinski original, es decir cuando  $n$  aumenta indefinidamente, también lo hace la cantidad de triángulos rectángulos, en cuanto el área de cada uno, van disminuyendo progresivamente en cada etapa, al igual que la sumatoria de todos estos

## Ejercicio 5

Realice la misma actividad para el triángulo modificado visto en el ejercicio 2

- **B. Complete la tabla según las etapas del triángulo de sierpinski:**

Etapas	Número de triángulos sombreados	Área de cada triángulo	Área total sombreada
0	1	1	1
1	6	1/9	6/9
2	36	1/81	36/81
3	216	1/729	216/729
4	1296	1/6561	1296/6561
5	7776	1/59049	7776/59049

- **C. ¿Cuál es la constante multiplicativa utilizada para encontrar el número de triángulos de una etapa a otra? ¿y en qué caso del área ?**

La constante utilizada para encontrar el número de triángulos remanentes en cada etapa es: “6” . En cuanto el área de cada triángulo, la constante es “1/9” porque se subdivide en 9 triángulos como tal

- **D. Encuentre las expresiones para calcular el número de triángulos. el del área de cada uno de ellos en la etapa  $n$  y el área total sombreada**

La expresión para hallar el número de triángulos remanentes en una etapa  $n$  es :  $6^n$

La expresión para hallar el área de cada triángulo en una etapa  $n$  es:  $\left(\frac{1}{9}\right)^n$

La expresión para hallar el total de área de todos los triángulos en una etapa  $n$  es:  $\left(\frac{6}{9}\right)^n$

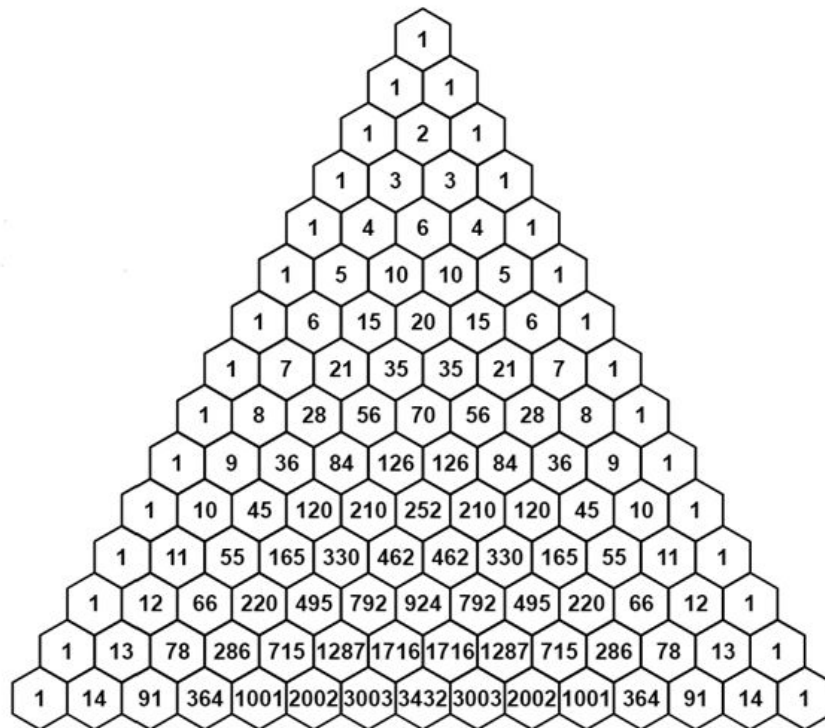
- **E. Cuando  $n$  aumenta indefinidamente, ¿Qué sucede con el número de triángulos? ¿que ocurre con el área de cada triángulo? ¿con el área total sombreada? ¿y con el área sin sombrear?**

Al igual que los casos anteriores cuando las etapas  $n$  van haciéndose mayores, el número de triángulos remanentes aumenta, en cuanto las áreas de estos van disminuyéndose progresivamente, al igual que la sumatoria de todas estas, ahora si nos referimos a los triángulos centrales, su área va creciendo a medida que se van aumentando las etapas

## Hoja de trabajo 3

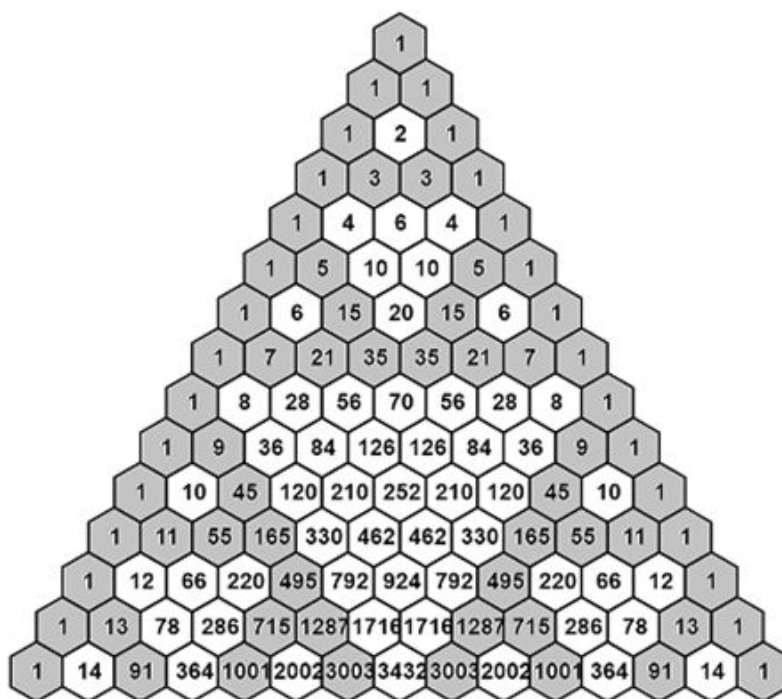
### Ejercicio 1

a. Completar el Triángulo de Pascal hasta la fila 15.



### Ejercicio 2

a. Colorear con negro las celdas impares.



**b. Observar la figura y responder:**

**b.1. ¿De qué color es el elemento que se encuentra debajo de dos negras?**

Respuesta: El color debajo del elemento que se encuentra debajo de dos negras es blanco.

**b.2. ¿De qué color es el elemento que se encuentra debajo de dos blancas?**

Respuesta: El color debajo del elemento que se encuentra debajo de dos blancas es blanco.

**b.3. Complete lo siguiente,**

negro + negro = blanco + blanco = blanco, de acuerdo a  $b_1, b_2$ .

**b.4. Y ¿blanco + negro?**

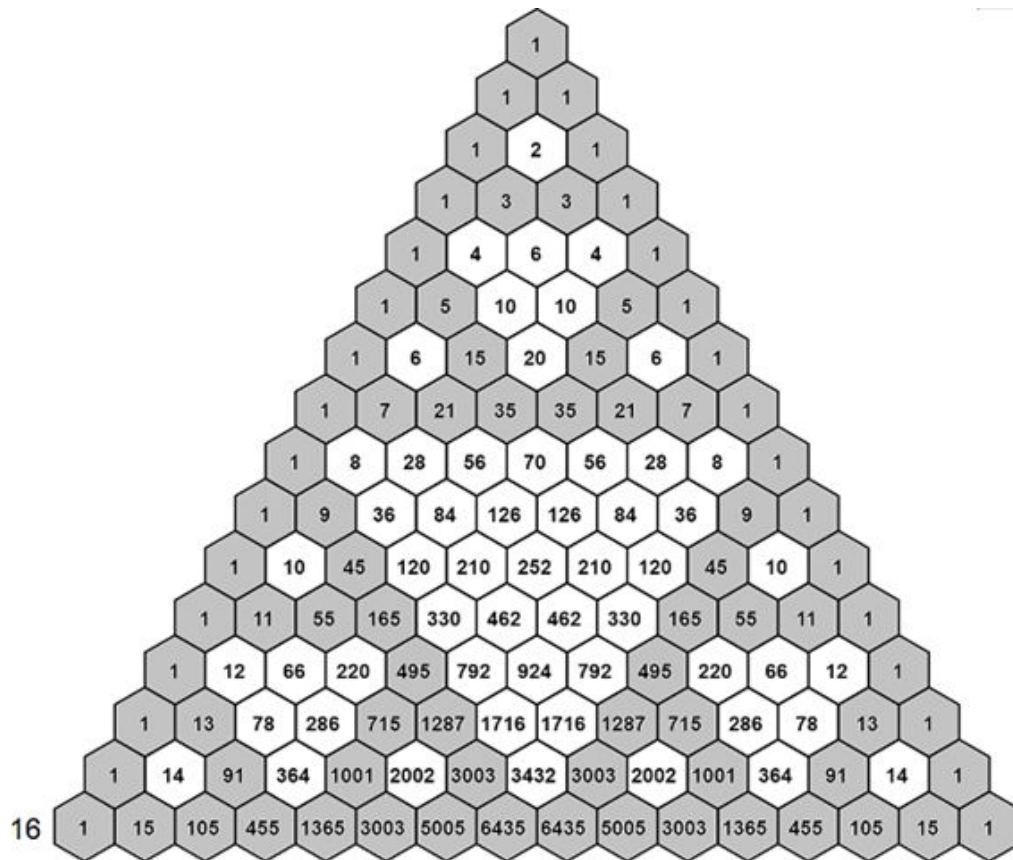
Blanco + negro = negro

**c. Enuncie una regla, en función de los colores, para pintar un elemento del renglón k-ésimo teniendo en cuenta los colores de los dos elementos inmediatamente superiores del renglón k-1-ésimo.**

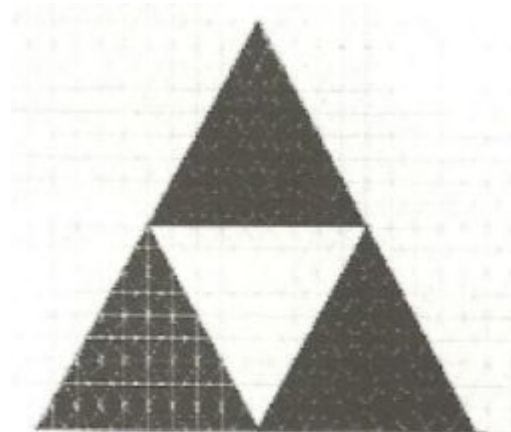
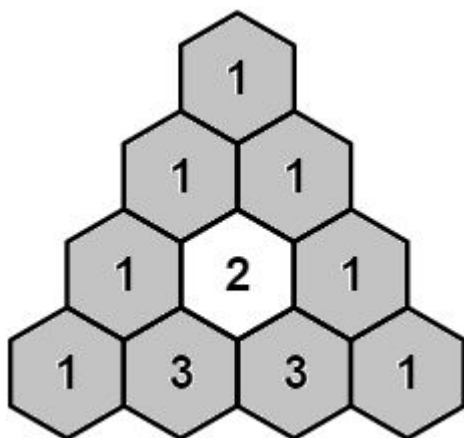
En el triángulo de Pascal, un elemento de un renglón k-ésimo será color blanco si los dos elementos inmediatamente superiores a él, en el renglón k-1-ésimo son del mismo color. Será color negro en otro caso.

**d. Colorear los elementos del renglón 16 usando el algoritmo visto en c.**

Observando la figura coloreada y aplicando el algoritmo descrito en el ítem c, es posible concluir que todos los elementos del renglón 16 serán negros dado que en el renglón 15 no hay dos elementos consecutivos de igual color. El renglón 16 entonces, estaría conformado por elementos impares de la siguiente forma

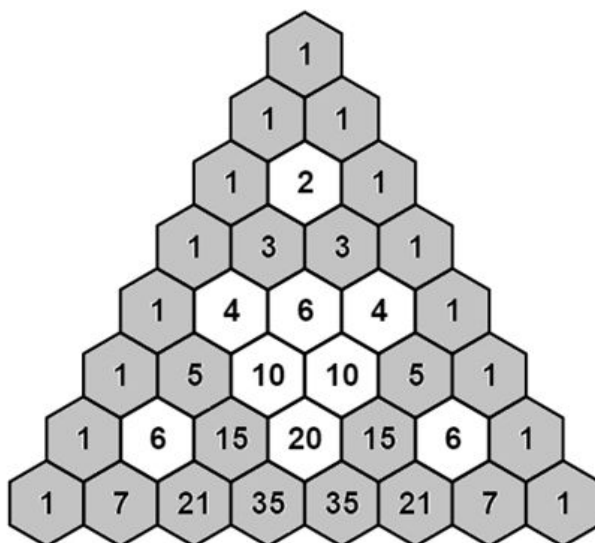


- e. Observe los primeros cuatro renglones pintados y compare con el triángulo de Sierpinski. ¿Puede encontrar alguna relación entre el triángulo de Pascal y el de Sierpinski?



Observando los primeros cuatro renglones del triángulo de pascal y la etapa 1 del triángulo de Sierpinski, podría verse al elemento blanco del triángulo de Pascal como si fuera el triángulo que se encuentra en el centro del triángulo de Sierpinski.

- f. Obsérvese los primeros ocho renglones ¿existe una relación con los cuatros primeros? ¿y con el triángulo de Sierpinski?



En los 8 primeros renglones coloreados del triángulo de Pascal, presentado en la Fig. 6, y comparándolos con los primeros 4 renglones de la Fig. 4 se observa que nuevamente se presenta una figura triangular en el centro, sin embargo, también aparecen dos elementos blancos a los costados.

Si ahora se compara la figura obtenida con el triángulo de Sierpinski de la etapa 2, la relación es clara, el patrón es igual en ambas.

- g. Si colorea los 8 renglones siguientes ¿qué etapa del triángulo de Sierpinski aparece?

Coloreando los siguientes 8 renglones se obtendría la Fig. 3 nuevamente, en ella se puede apreciar el triángulo de Sierpinski de la etapa 3.

- h. ¿Cuántos renglones se deben colorear para representar la etapa 4 del triángulo de Sierpinski?, ¿y la etapa 5?

Habiendo analizando los anteriores casos, la conclusión es que el triángulo de Pascal representará una etapa del triángulo de Sierpinski siguiendo la siguiente ecuación:

$$N^{\circ} \text{ renglones} = 2^{n+1}$$

Donde  $n$  es la etapa del triángulo de Sierpinski que se quiere representar.



Mediante la fórmula es posible calcular los renglones necesarios para una determinada etapa de Sierpinski. Entonces, para hallar las etapas 4 y 5 del triángulo de Sierpinski sería necesarios los siguientes renglones:

$$\text{Etapa 0} = 2^{0+1} = 2 \text{ renglones}$$

$$\text{Etapa 1} = 2^{1+1} = 4 \text{ renglones}$$

$$\text{Etapa 2} = 2^{2+1} = 8 \text{ renglones}$$

$$\text{Etapa 3} = 2^{3+1} = 16 \text{ renglones}$$

$$\text{Etapa 4} = 2^{4+1} = 32 \text{ renglones}$$

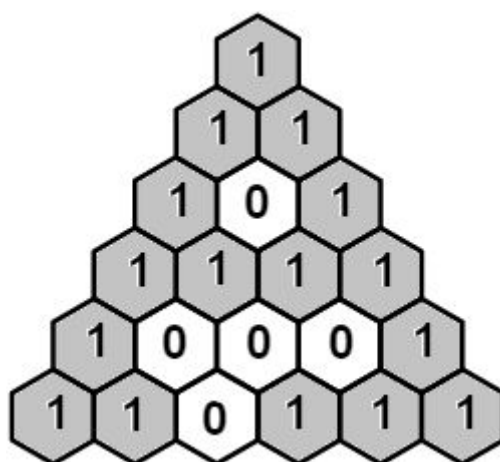
$$\text{Etapa 5} = 2^{5+1} = 64 \text{ renglones}$$

- i. ¿Es cierta la siguiente afirmación?: en el triángulo de Pascal se tienen tantos números pares como impares?

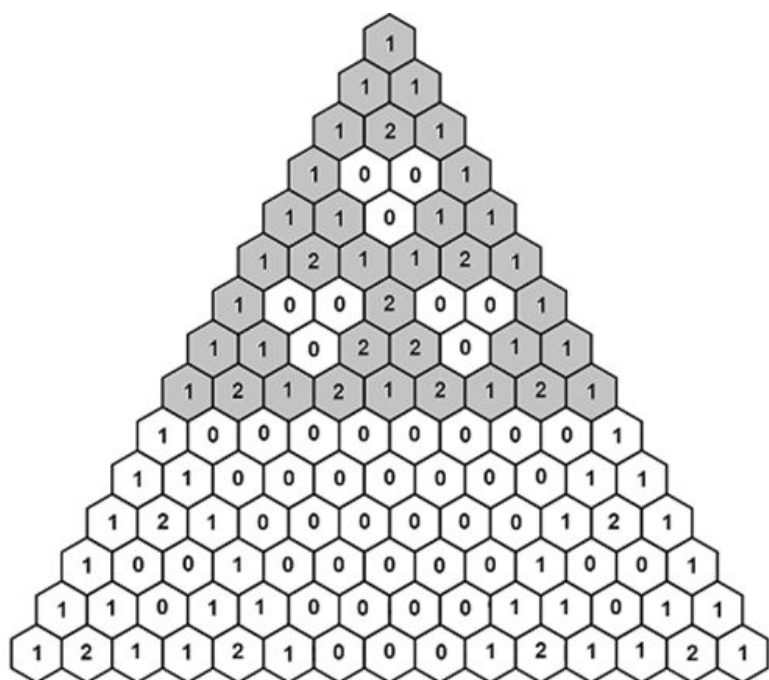
La afirmación no es cierta, la relación entre números pares e impares varía dependiendo con cuántos renglones se trabaje. Por ejemplo, observando la Fig. 4, se aprecian nueve elementos impares y sólo uno par. En la Fig.6 esto se repite, hay mayor número de elementos impares, sin embargo, al analizar el triángulo de Pascal de 32 renglones, el número de elementos impares será menor.

#### Aritmética módulo 2:

A los números pares se les asigna el valor 0 porque es el resto que poseen al ser divididos por 2, y a los impares se le asigna el valor 1 ya que al ser divididos por 2 presentan ese resto. Coloreando los números impares:

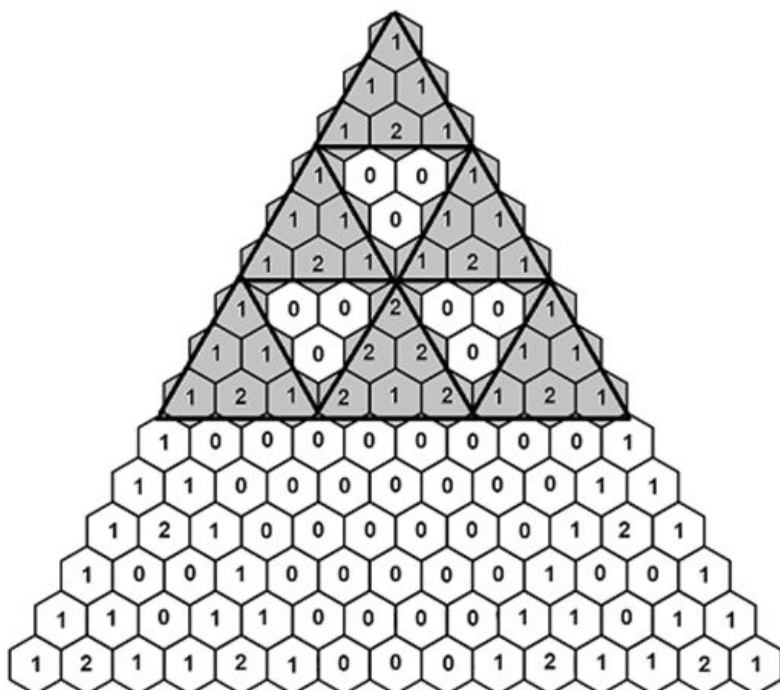


- j. Completar los 15 primeros renglones del triángulo de Pascal según la aritmética módulo 3 y colorear los 9 primeros renglones, según la siguiente regla: Si la entrada es 1 o 2 pinte la casilla de negro. Si la entrada es 0 deje blanco.

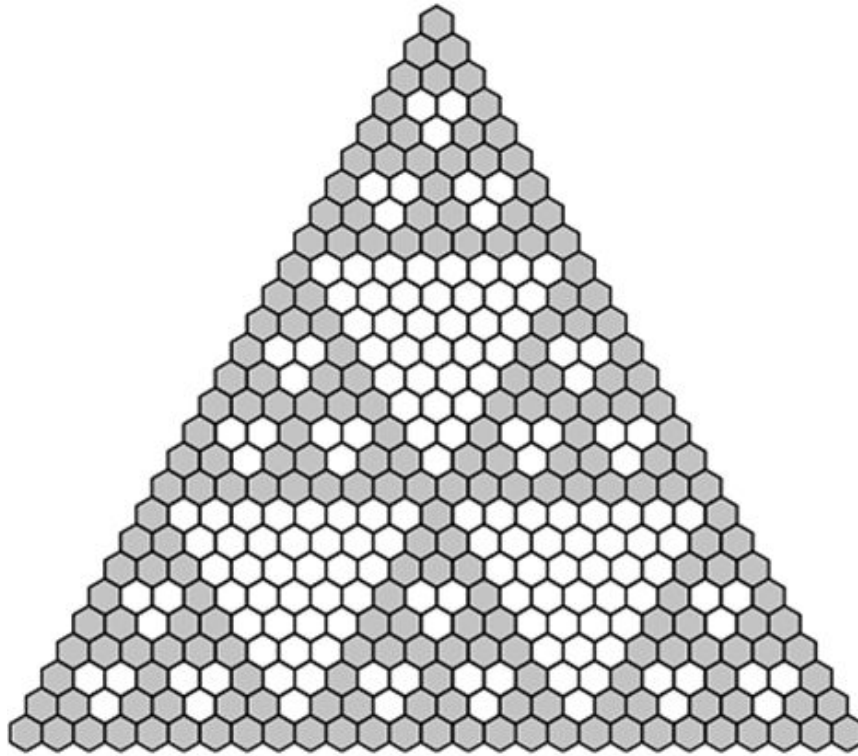


k. Comparar la figura obtenida en los 9 primeros renglones con el triángulo de Sierpinski modificado.

Al observar la del ítem j) y compararla con el triángulo de Sierpinski modificado se puede apreciar claramente la similitud entre ambos. Los elementos blancos forman tres triángulos invertidos y los elementos coloreados seis triángulos, su posición y orientación se corresponde con los triángulos en el triángulo de Sierpinski.



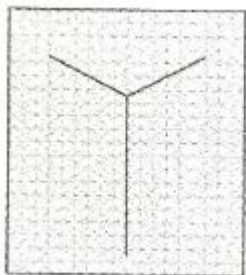
- I. Colorear las 18 filas siguientes según el algoritmo anterior y comparar, nuevamente, con el triángulo de Sierpinski modificado.**



Con 27 renglones, el triángulo de Pascal obtenido en el ítem **j)** se asemeja al triángulo de Sierpinski modificado de la etapa 2.

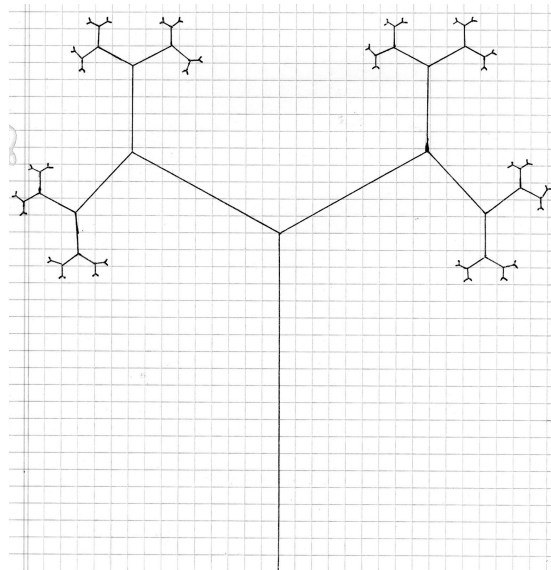
## Hoja de trabajo 4

### Ejercicio 1



- **A.** En la figura se ha dibujado la etapa 1 del árbol. Dibuje las 4 ramas nuevas para la etapa 2.

Dibuje ocho nuevas ramas para la etapa 3 y repita nuevamente para la etapa 4.



Continúe el proceso hasta que las ramas sean muy pequeñas para ser dibujadas.

Supongamos que la longitud inicial del árbol sea igual a 1. La longitud de cada una de las dos primeras ramas es  $\frac{1}{2}$ .

- **B.** ¿Cuántas ramas tienen longitud  $\frac{1}{4}$ ? ¿Y  $\frac{1}{16}$ ?

Partiendo de que el nivel 1 tiene longitud 1 tendremos que, la longitud de los siguientes niveles será:

$$\frac{\text{Longitud anterior}}{2}$$

Por otro lado la cantidad de ramas es duplicada en cada nivel:

$$\text{cantidad anterior} * 2$$

Resultando que el número de ramas es igual al denominador de valor obtenido en la longitud, por lo tanto, el árbol contiene 4 ramas de longitud  $\frac{1}{4}$  y 16 de longitud  $\frac{1}{16}$ .

- **C.** ¿Cuánto suman las longitudes de todas las ramas de longitud  $\frac{1}{4}$ ? ¿Y  $\frac{1}{16}$ ?

Al realizar la suma de las longitudes de las ramas de cada nivel obtenemos el valor del nivel inicial, en este caso 1.

$$\frac{\text{Longitud anterior}}{2} * (\text{cantidad anterior} * 2)$$

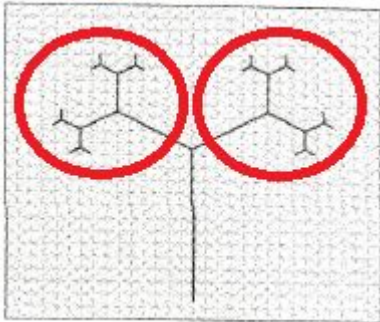
$$\frac{1}{2} * (2 * 2) = 1$$

$$\frac{1}{8} * (8 * 2) = 1$$

- **D.** Si repitiéramos el algoritmo  $n$  veces, ¿cuál sería el valor de la suma de las longitudes de todas las ramas del árbol?

El valor de aumento en la cantidad de ramas es equivalente al de reducción de longitud, por lo tanto el valor de la suma de las longitudes de las ramas de cada nivel será el valor del nivel inicial, en este caso 1 y, a medida que avancen los niveles, la longitud va ir tendiendo a cero.

- **E.** ¿Puedo encontrar partes del árbol que se parezcan al árbol entero?

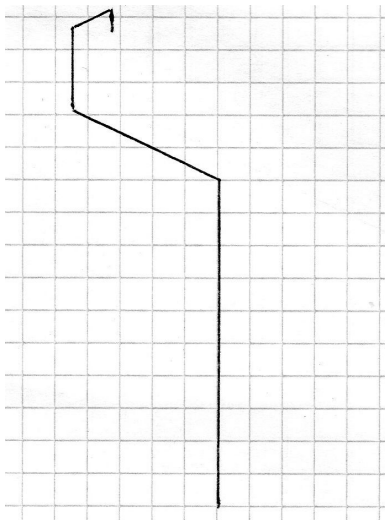


Las 2 ramas marcadas presentan un parecido con el árbol entero, aunque le falta un nivel para que fueran perfectamente iguales.

- **F.** Dibuje en la parte inferior de la hoja y en medio de un segmento al que le asignaremos longitud 1. Tire la moneda una vez; si sale cara, dibuje un segmento de longitud  $\frac{1}{2}$  hacia la derecha, si sale seca, dibujelo hacia la izquierda.

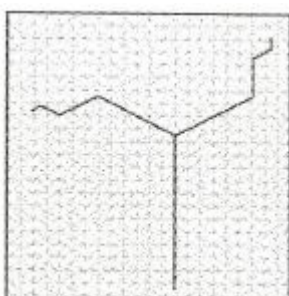
Vuelva a tirar el dado, si sale cara, vuelva a dibujar un segmento, ahora de longitud  $\frac{1}{4}$  hacia la derecha; si sale seca, dibujelo hacia la izquierda.

Una secuencia puede ser: derecha, derecha, izquierda, izquierda... (DDII) o izquierda, izquierda, derecha, derecha (IIDD), etc.



La secuencia obtenida fue: Izquierda (seca) - Derecha (cara) - Derecha (cara) - Derecha (cara)

- **G.** Indique qué secuencia sigue la rama de la izquierda y la rama de la derecha del árbol siguiente:



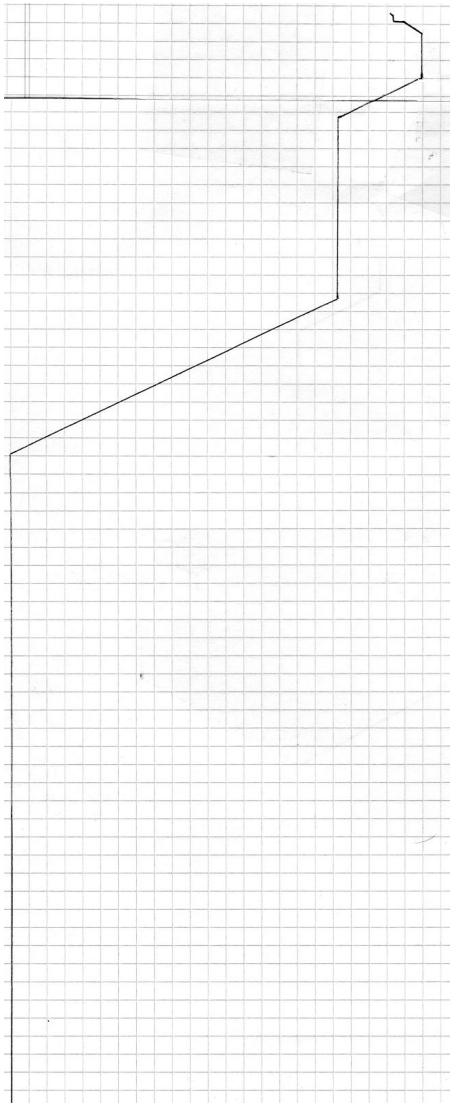
La rama de la izquierda sigue la secuencia:

Izquierda (seca) - Izquierda (seca) - Derecha (cara) - Izquierda (seca)

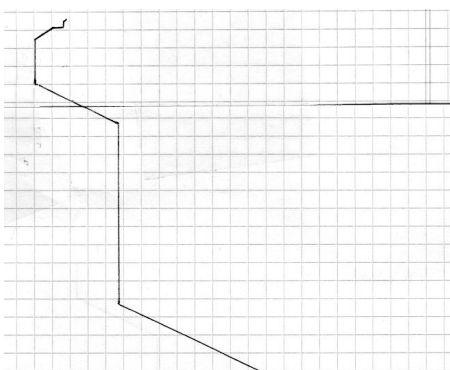
La rama de la derecha sigue la secuencia:

Derecha (cara) - Izquierda (seca) - Derecha (cara) - Izquierda (seca)

- **H.** Dibuje la rama del árbol que sigue la siguiente secuencia: DIDIIDI.



- **I.** Trace la rama simétrica a la anterior. ¿Cuál es la secuencia?



La secuencia de la simétrica es:

Izquierda - Derecha - Izquierda - Derecha - Derecha - Derecha - Izquierda - Derecha

- **J.** ¿Cómo debería ser la secuencia para que la ramificación formará una espiral?

Enuncie por lo menos 2 posibilidades.

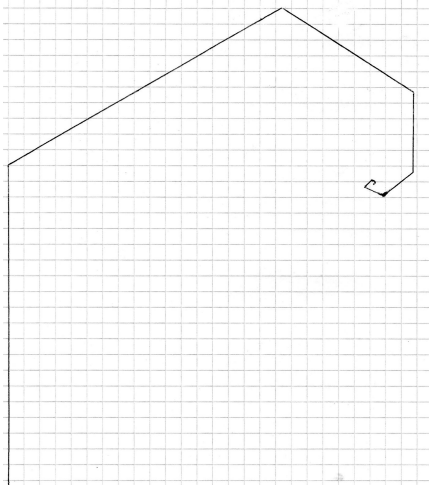
Para que se forme un espiral la secuencia debería ser siempre al mismo lado:

Derecha - Derecha - Derecha - Derecha - Derecha - Derecha...

Izquierda - Izquierda - Izquierda - Izquierda - Izquierda - Izquierda...

## Ejercicio 2

- A. Dibuje la poligonal comenzando por el tronco y girando siempre a la derecha.



- **B.** Compruebe que la longitud de la espiral hallada en el ítem a es finita.

La longitud del espiral va a estar dado por:

$$L = 1 + \sum_{i=1}^n \frac{1}{2^i} \quad \text{Donde n representa el nivel de la rama.}$$

Si tomando la sumatoria, al realizar las sumas sucesivas con un valor de  $n=34$  obtenemos:

$$\sum_{n=1}^{34} \frac{1}{2^n} = \frac{2^{34}-1}{2^{34}} = 0,9999999$$

A partir del valor de  $n=35$ , comenzaremos a obtener valores igual a 1

$$\sum_{n=1}^{35} \frac{1}{2^n} = \frac{2^{35} - 1}{2^{35}} = 1$$

$$\sum_{n=1}^{50} \frac{1}{2^n} = \frac{2^{50} - 1}{2^{50}} = 1$$

Esto comprobará que en el nivel 35 se alcanza la longitud 2 (sumando el tronco) y esta longitud se mantiene en los siguientes niveles.



- **C.** ¿Cuántas espirales de longitud 2 se pueden dibujar?

Teniendo en cuenta la explicación anterior, podemos decir que la cantidad de espirales de longitud 2 que tendremos será igual a:

$$n - 34$$

Ya que hasta el nivel 34 no se alcanza la longitud 2.

- **D.** Encontrar 4 espirales de longitud 1. En el árbol, ¿cuántas espirales de longitud  $\frac{1}{2}$  se pueden encontrar?

Sin tomar en cuenta al tronco, partiendo desde las primeras ramas, alcanzaremos la longitud 1 a partir del nivel 35, por lo tanto a partir de este nivel la longitud será 1:

$$\sum_{n=1}^{35} \frac{1}{2^n} = \frac{2^{35} - 1}{2^{35}}$$

$$\sum_{n=1}^{36} \frac{1}{2^n} = \frac{2^{36} - 1}{2^{36}}$$

$$\sum_{n=1}^{37} \frac{1}{2^n} = \frac{2^{37} - 1}{2^{37}}$$

$$\sum_{n=1}^{38} \frac{1}{2^n} = \frac{2^{38} - 1}{2^{38}}$$

Para los espirales de longitud  $\frac{1}{2}$  no se tomarían el tronco y la rama del primer nivel que tiene longitud  $\frac{1}{2}$ , por lo tanto la cantidad de ramas de longitud  $\frac{1}{2}$  serán:

$$n - 33$$

Ya que durante 33 niveles no alcanzaremos la longitud  $\frac{1}{2}$ .

- **E.** Si colecciona todas las espirales que puedan encontrar en el árbol se puede observar, claramente, que no son de igual longitud. ¿Se puede decir que estas ramas son réplicas una de otra excepto por la longitud?

Podemos decir que los espirales obtenidos serán réplicas, salvando la diferencia de longitud.

- **F.** ¿Cual es la longitud de una rama completa? ¿Y si incluimos el tronco?

La longitud de una rama completa es 1 y tomando en cuenta el tronco la longitud sería 2.

- **G.** Escriba la sucesión generada al considerar las longitudes de las distintas ramas.

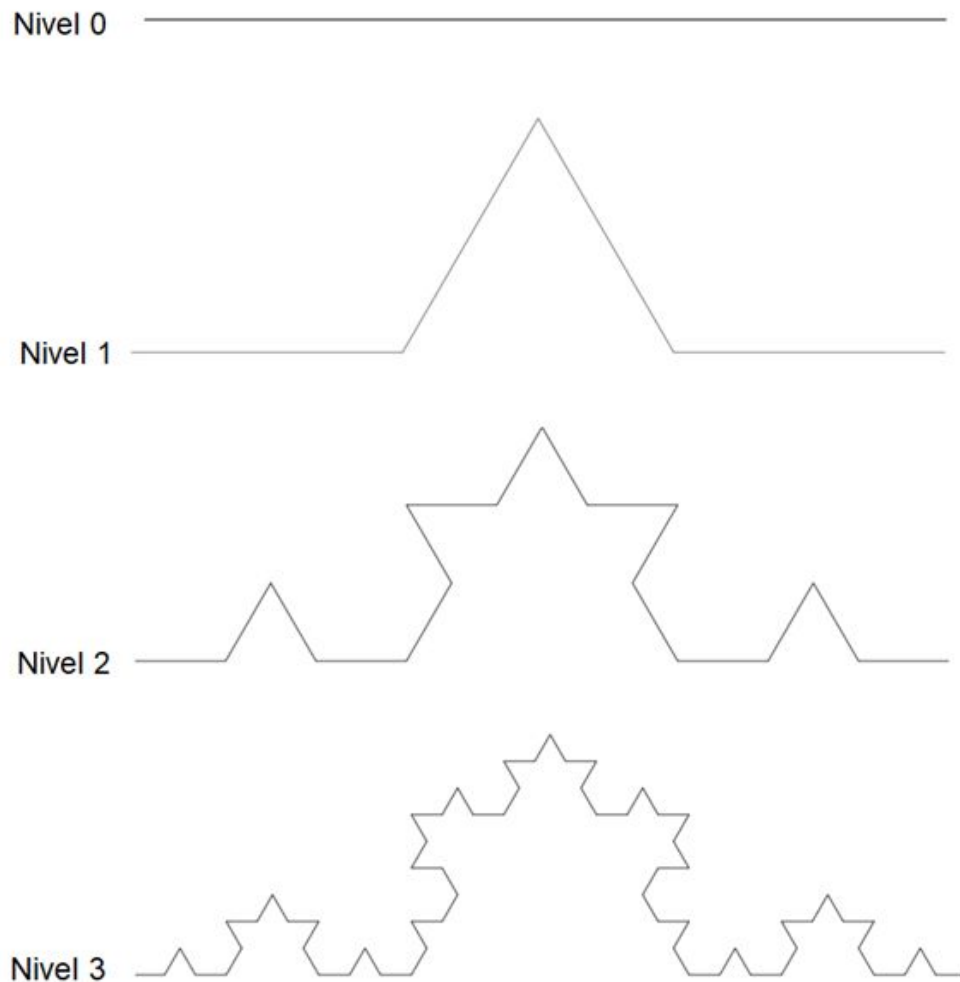
La sucesión que calculará la longitud de las ramas es:

$$\sum_{n=1}^m \frac{1}{2^n} \quad \text{Donde m será los niveles que tenga el árbol.}$$

# Hoja de trabajo 5

## Ejercicio 1

a. Partiendo del segmento de longitud 1 repita el algoritmo tres veces.



b. Cuento el número de segmentos en cada etapa y encuentre la ley que siguen. Haciendo uso de la misma, calcule el número para la etapa 4 y 5 y, posteriormente, complete la siguiente tabla.

- La etapa 0 consiste de 1 segmento.
- La etapa 1 consiste de 4 segmentos.
- La etapa 2 consiste de 16 segmentos.
- La etapa 3 consiste de 64 segmentos.

Analizando el número de segmentos obtenidos en cada una de las cuatro primeras etapas de la curva de Koch, es posible deducir que el número de segmentos que una determinada etapa tendrá se puede expresar como

$$\text{Nro segmentos} = 4^n$$

donde n es la etapa. Aplicando la fórmula para completar la tabla

Etapa	N° de segmentos
0	1
1	4
2	16
3	64
4	256
5	1024
...	...
n	$4^n$

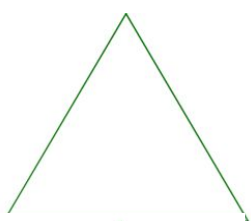
- c. Imagine que el proceso se repite de manera indefinida. Describa cómo evoluciona la poligonal. La curva obtenida es la llamada curva de Koch.

Cuando la curva de Koch es sometida a infinitas iteraciones, aumenta infinitamente la cantidad de segmentos por lo tanto también su perímetro se vuelve infinito. Conforme aumenta el número de iteraciones, se vuelve difícil identificar cada uno de los triángulos y toma la forma de una curva suave (aunque en realidad no lo es).

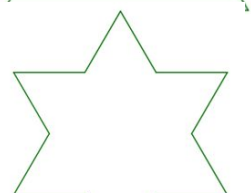
## Ejercicio 2

- a. Construya el fractal copo de nieve hasta la etapa 3.

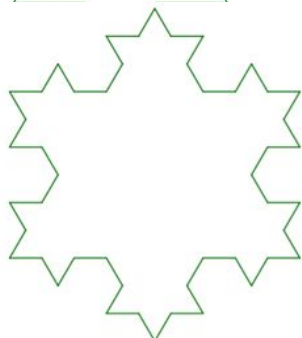
**Etapa 0**



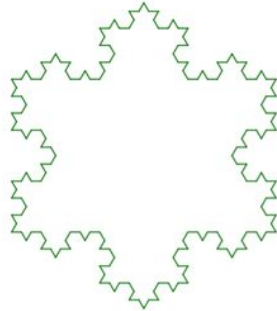
**Etapa 1**



**Etapa 2**



### Etapa 3



b. Cuente el número de segmentos que se forman en cada etapa y complete la tabla 2. Infiera qué ocurre en las etapas 4, 5 y n.

- La etapa 0 consiste de 3 segmentos.
- La etapa 1 consiste de 12 segmentos.
- La etapa 2 consiste de 48 segmentos.
- La etapa 3 consiste de 192 segmentos.

Etapas	N° de segmentos	Longitud de cada segmento
0	3	1
1	12	1/3
2	48	1/9
3	192	1/27
4	768	1/81
5	3072	1/243
...	...	...
n	$\text{Etapa}(n-1) * 4$	$1/3^n$

Considerando que el segmento inicial tiene longitud 1, la expresión para obtener la longitud de los segmentos de las n etapas es:

$$\frac{1}{3^n}$$

Generalizando para segmentos de longitud distinta de 1

$$\frac{L_0}{3^n}$$

donde  $L_0$  es la longitud inicial de cada uno de los tres segmentos.

- c. Encuentre la constante multiplicativa que permite calcular el número de segmentos de una etapa a otra.

Observando los resultados obtenidos, la constante multiplicativa para hallar el número de segmentos de una etapa  $n$ , es 4. Sabiendo que en la etapa 0 el copo de nieve posee 3 segmentos, se puede calcular el número de segmentos de cualquier etapa  $n$  aplicando la siguiente fórmula

$$\text{Nro segmentos etapa}_n = \text{Nro segmentos etapa}_{n-1} * 4$$

- d. Si el proceso se repite indefinidamente indique cómo cambia la figura.

En cada etapa del copo de nieve, a cada segmento se lo divide en 3 partes iguales y se obtienen 4 segmentos de igual longitud, por lo tanto, conforme aumenten las iteraciones la longitud de cada segmento disminuirá, pero el número de ellos aumentará, el perímetro de esta figura será infinito.

### Ejercicio 3

- a. De una etapa a otra el perímetro de la figura aumenta. Calcule el perímetro en las cinco primeras etapas y generalice para la etapa  $n$ . Complete la tabla 3.

Etapas	longitud del perímetro
0	3
1	4
2	16/3
3	64/9
4	256/27
5	1024/81
...	...
$n$	$3 * (\frac{4}{3})^n$

- b. Encuentre la constante multiplicativa para calcular el perímetro de una etapa a otra.

Sabiendo que en la etapa 0 se tienen 3 segmentos de igual longitud, y que en cada etapa a cada uno de esos segmentos se los divide en 3 y se agrega un triángulo sin base en el segmento central para dar lugar a 4 segmentos, se infiere que la longitud del perímetro aumenta en un factor de  $4/3$ , la constante multiplicativa.

$$\text{longitud de perímetro etapa } n = 3 * (\frac{4}{3})^n$$

- c. Si el proceso iterativo continúa indefinidamente, ¿cuál será el perímetro del fractal copo de nieve resultante?

Dado que  $4/3$  es un número mayor que 1, cada vez que se multiplica  $4/3$  por la longitud anterior, el perímetro aumenta. La conclusión es que cuanto mayor sea el número de iteraciones mayor será el perímetro, si las iteraciones son infinitas, el perímetro también lo será.

## Ejercicio 4

- a. A medida que el copo de nieve crece su área también lo hace. Calcule el área para las etapas 3, 4 y 5 y complete la siguiente tabla:

Etapas	área
0	$\frac{\sqrt{3}}{4}$
1	$\frac{\sqrt{3}}{4} + \frac{\sqrt{3}}{4} \cdot \frac{3}{9}$
2	$\frac{\sqrt{3}}{4} + \frac{\sqrt{3}}{4} \cdot \frac{3}{9} + \frac{\sqrt{3}}{4} \cdot \frac{4}{9^2} = \frac{\sqrt{3}}{4} \left( 1 + \frac{3}{9} + 3 \cdot \frac{4}{9^2} \right)$
3	$\frac{\sqrt{3}}{4} + \frac{\sqrt{3}}{4} \left( \frac{3}{9} + 3 \cdot \frac{4^1}{9^2} + 3 \cdot \frac{4^2}{9^3} \right)$
4	$\frac{\sqrt{3}}{4} + \frac{\sqrt{3}}{4} \left( \frac{3}{9} + 3 \cdot \frac{4^1}{9^2} + 3 \cdot \frac{4^2}{9^3} + 3 \cdot \frac{4^3}{9^4} \right)$
5	$\frac{\sqrt{3}}{4} + \frac{\sqrt{3}}{4} \left( \frac{3}{9} + 3 \cdot \frac{4^1}{9^2} + 3 \cdot \frac{4^2}{9^3} + 3 \cdot \frac{4^3}{9^4} + 3 \cdot \frac{4^4}{9^5} \right)$
...	...
n	$\frac{\sqrt{3}}{4} + \frac{\sqrt{3}}{4} \sum_{i=1}^n 3 \cdot \frac{4^{i-1}}{9^i}$

- b. Encuentre la expresión del área para el nivel n

El área de un triángulo equilátero viene dada por la expresión

$$\frac{\sqrt{3} * l^2}{4}$$

donde  $l$  es la longitud de los lados. Si la longitud inicial de sus lados es 1 y en la primera iteración se añaden 3 triángulos y 4 nuevos segmentos por cada lado que reducen su longitud en  $1/3$ , el área luego de una iteración será el área de la etapa 0 más el área de los nuevos triángulos

$$A_1 = \frac{\sqrt{3}}{4} + \frac{\sqrt{3}}{4} * 3 \left( \frac{1}{3} \right)^2$$

Para n iteraciones se tiene

$$A_n = \frac{\sqrt{3}}{4} + \frac{\sqrt{3}}{4} \sum_{1}^n 3 * \frac{4^{n-1}}{9^n}$$

$$A_n = \frac{\sqrt{3}}{4} + \frac{\sqrt{3}}{4} \left( \frac{1}{9} \sum_{1}^n 3 * \left( \frac{4}{9} \right)^{n-1} \right)$$

c. ¿Cuál es el área del copo de nieve completo?

Al realizar infinitas iteraciones

$$\lim_{n \rightarrow \infty} A_n = \lim_{n \rightarrow \infty} \frac{\sqrt{3}}{4} + \left( \frac{\sqrt{3}}{4} \frac{1}{9} \sum_{1}^n 3 * \left( \frac{4}{9} \right)^{n-1} \right) = \frac{\sqrt{3}}{4} \lim_{n \rightarrow \infty} \left( 1 + \frac{1}{9} \sum_{1}^n \left( 3 * \left( \frac{4}{9} \right)^{n-1} \right) \right)$$

$$A_{total} = \frac{\sqrt{3}}{4} + \frac{\sqrt{3}}{4} * \frac{3}{5} = \frac{2\sqrt{3}}{5}$$

d. ¿Es el perímetro del fractal copo de nieve finito o infinito? ¿Es el área finita o infinita?

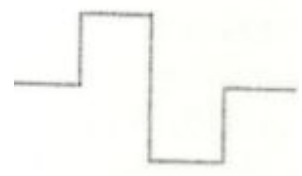
El perímetro es infinito

$$\lim_{n \rightarrow \infty} Perimetro = \lim_{n \rightarrow \infty} 3 * \left( \frac{4}{3} \right)^n = \infty$$

El área fue calculada en el ítem anterior, es  $\frac{2\sqrt{3}}{5}$ , finita.

## Ejercicio 5

A partir de la construcción del fractal copo de nieve resulta sencillo generar otros fractales siguiendo una ley de recurrencia; por ejemplo, si en vez de dividir el segmento unidad en tres partes iguales, se lo divide en cuatro partes iguales y se construye un cuadrado hacia arriba en el segmento 2, eliminando la base, y un cuadrado hacia abajo en el segmento 3, eliminando el segmento paralelo a la base.





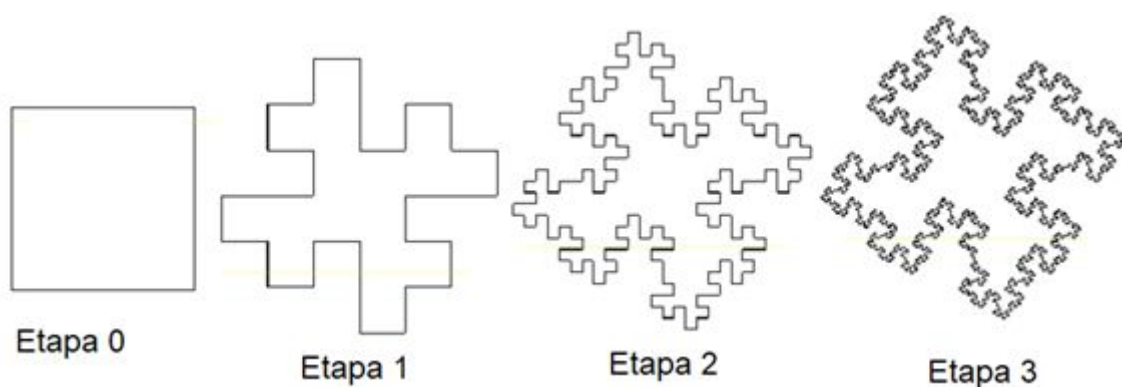
- a. La poligonal obtenida, a la que llamaremos C1 tiene 8 lados de longitud  $\frac{1}{4}$  y por lo tanto su longitud total será  $\frac{1}{4} * 8 = 2$ . Si se repite la operación sobre cada uno de estos lados resulta una poligonal C2 de 64 lados, cada uno de longitud  $\frac{1}{16}$ .
- b. Proceda de igual modo para C3, C4 y C5 y complete la Tabla 5.

Etapas	N° de segmentos	Longitud de cada segmento	Longitud total
0	1	1	1
1	8	$\frac{1}{4}$	2
2	64	$\frac{1}{16}$	4
3	512	$\frac{1}{64}$	8
4	4096	$\frac{1}{256}$	16
5	32768	$\frac{1}{1024}$	32
...	...	...	...
n	$8^n$	$\frac{1}{4^n}$	$2^n$

- c. ¿Qué ocurre cuando n aumenta indefinidamente?

Observando la tabla del ítem anterior, cuando n aumenta indefinidamente el número de segmentos aumenta, pero su longitud disminuye. El perímetro total tiende a infinito igual que en casos anteriores.

- d. Dado un cuadrado, cuyo lado tiene longitud 1. Si se repite este algoritmo sobre cada lado del cuadrado, indefinidamente, se obtiene una curva fractal cerrada. Realice la iteración de la gráfica, hasta la etapa 3.



- e. Complete la siguiente tabla:

Etapas	Longitud del perímetro	Valor del área
0	4	1
1	8	1
2	16	1
3	32	1
4	64	1
5	128	1
...	...	...
n	$4 * 2^n$	1

f. ¿Qué ocurre con el perímetro cuando n aumenta indefinidamente?, ¿y con el área?

Nuevamente, observando la tabla del **ítem e)** y teniendo en cuenta los valores obtenidos en el ítem 5.a, se encuentra que el perímetro aumenta conforme aumentan las iteraciones mientras que el área se mantiene constante y es 1, que es la longitud del lado del cuadrado inicial.

# Hoja de trabajo 6

## Ejercicio 1

- A. ¿Qué puede decir usted en cuanto a la autosemejanza de un gajo de coliflor con respecto a la coliflor completa? ¿Y del brócoli? ¿Qué otros objetos autosemejantes se pueden encontrar en la naturaleza?

Tanto un gajo de coliflor como uno de brócoli tendrán autosemejanza con el completo pero, esta autosimilaridad, no sería estrictamente autosimilar ya que las partes en las que se descomponen no serán réplicas exactas al completo.

Entre los ejemplos de autosemejanza que pueden verse en la naturaleza podemos mencionar:

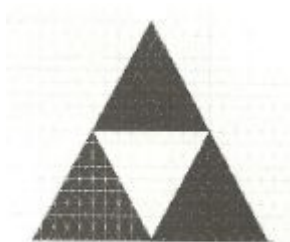
- Rayo.
- Árbol.
- Sistema circulatorio.
- Nubes.
- Montañas.

- B. Explique por que solo el árbol completo es estrictamente autosemejante.

Se dice que hay autosimilitud exacta cuando una o varias partes de un todo repiten exactamente su similitud con ese todo. Dado que el árbol no crece de manera uniforme la autosimilitud no se presenta hasta que se haya desarrollado por completo.

- C. Considere el triángulo de Sierpinski y seleccione cualquier porción de cualquier tamaño que contenga una parte sombreada, ¿será una réplica exacta del triángulo completo? ¿Es el triángulo de Sierpinski estrictamente autosemejante?

¿Puede mencionar que otros fractales de los ya vistos cumplen con esta propiedad?



En esta imagen podemos notar que es posible extraer 3 réplicas exactas en una escala menor del triángulo original.

El Triángulo de Sierpinski es uno de los claros ejemplos que permiten observar la autosimilaridad exacta.

Otros ejemplos de autosimilaridad exacta pueden ser:

Alfombra (Tapiz) de Sierpinski, Conjunto de Cantor, Curva de Von Koch.

# Hoja de trabajo 7

## Ejercicio 1

Calcule la dimensión de los siguientes fractales:

- **A.** Conjunto del tercio medio de Cantor.

$$D = \frac{\ln(2)}{\ln(3)} = 0,6309297536$$

- **B.** Conjunto del quinto medio de Cantor.

$$D = \frac{\ln(4)}{\ln(5)} = 0,8613531161$$

- **C.** Triángulo y Carpeta de Sierpinski.

Triángulo de Sierpinski:

$$D = \frac{\ln(3)}{\ln(2)} = 1,584962501$$

Alfombra (Tapiz) de Sierpinski:

$$D = \frac{\ln(8)}{\ln(3)} = 1,892789261$$

- **D.** Curva de von Koch y del Copo de Nieve.

$$D = \frac{\ln(4)}{\ln(3)} = 1,261859507$$

- **E.** Dragón.

$$D = \frac{\ln(2)}{\ln(\sqrt{2})} = 2$$

# VIDEO

[https://frreutn-my.sharepoint.com/:v/g/personal/eliaszalegas\\_ca\\_frre\\_utn\\_edu\\_ar/EWiF-YgnBVBoT1n6VapFGMBcLr42HI-VQf4VJs8N4yqQ?e=RxbQWU](https://frreutn-my.sharepoint.com/:v/g/personal/eliaszalegas_ca_frre_utn_edu_ar/EWiF-YgnBVBoT1n6VapFGMBcLr42HI-VQf4VJs8N4yqQ?e=RxbQWU)

## CONCLUSIONES

A través del informe desarrollado, hemos podido aprender y comprender los conceptos fundamentales de los fractales.

Estudiamos diferentes formas de representar fractales, vimos el concepto de autosemejanza y el cálculo de dimensiones, a través de las diferentes hojas de ejercicios prácticos realizados, sacados del libro “Fractales un universo poco frecuentado” de Lina Monica Maria Oviedo.

También pudimos profundizar en el entendimiento de lo que serían las distintas aplicaciones de los fractales en la informática, abordando tanto los conceptos y beneficios de la compresión fractal de imágenes y generación por procedimiento.

## REFERENCIAS

<https://fractalescio.wordpress.com/2014/11/12/cuanto-mide-la-costa-de-gran-bretana/>

<https://www.geogebra.org/m/tUD6vpFr>

<https://www.youtube.com/watch?v=0cq7Hsunr10>

[https://es.wikipedia.org/wiki/Sistema\\_iterativo\\_de\\_funciones](https://es.wikipedia.org/wiki/Sistema_iterativo_de_funciones)

<http://ijarcet.org/wp-content/uploads/IJARCET-VOL-6-ISSUE-8-1164-1177.pdf>

<https://slideplayer.com/slide/5025176/>

<https://www.on1.com/products/resize/>

<http://www.ijstr.org/final-print/july2013/An-Exploration-Of-The-Generalized-Cantor-Set.pdf>