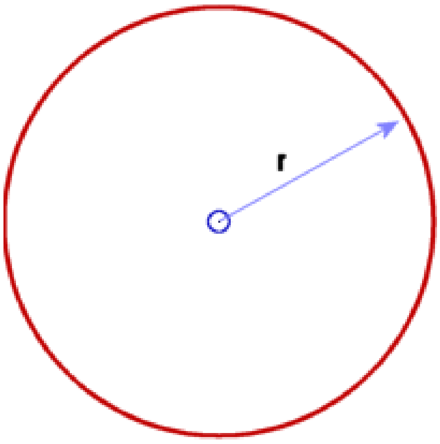
TRIÁNGULO DE SIERPINSKI. PROGRAMACIÓN EN PHP PDF

PROCESO DE CONSTRUCCIÓN

ITERACIÓN 0

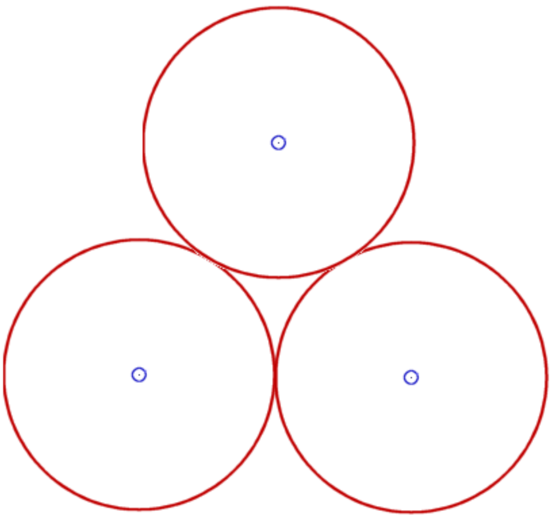
Partimos de un círculo de radio r.



En esta iteración, que llamamos 0 por ser el punto de inicio, tendremos un número de círculos $n_0 = 1$

ITERACIÓN 1

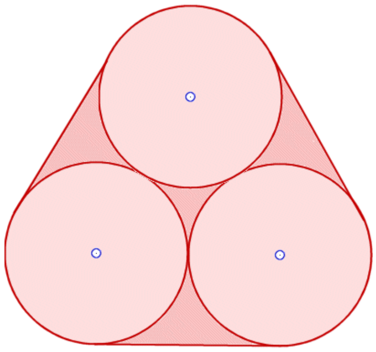
Con tres círculos iguales, los montamos tangencialmente de modo que sus centros formen un triángulo equilátero:



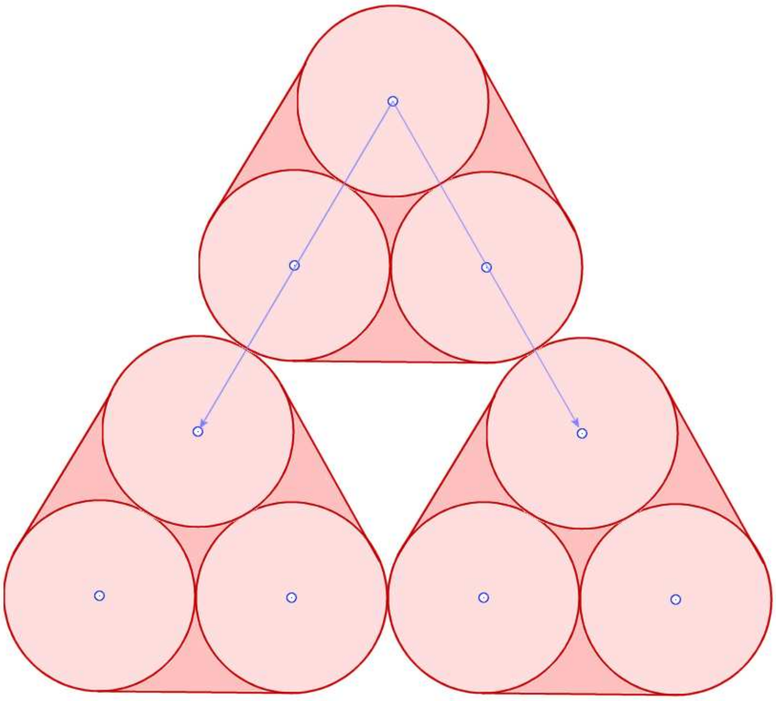
En esta iteración, tendremos un número de círculos $n_1 = 3 \cdot n_0 = 3$

ITERACIÓN K

Conforme vamos aumentando las iteraciones, observamos que toda la estructura de la iteración anterior se repite tres veces de modo similar. Si consideramos una iteración como una estructura encerrada en un "triángulo equilátero":

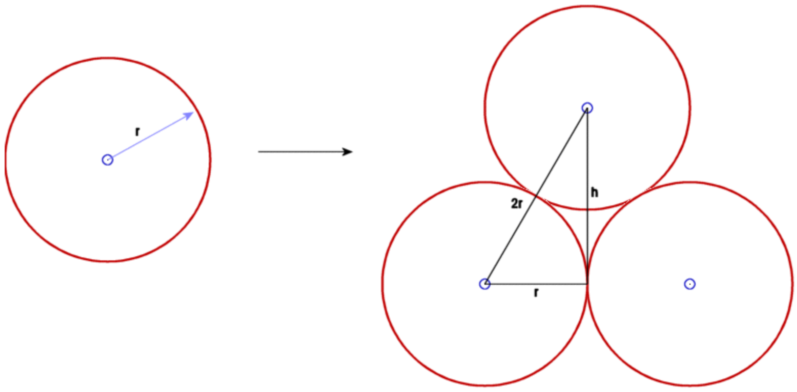


Bastaría con repetir esta estructura tres veces. A cada estructura la llamaremos BLOQUE:



Para ello, vamos a controlar los desplazamientos de cada bloque, tanto en horizontal (eje de abscisas) como en vertical (eje de ordenadas).

En la primera iteración:



$$h_1 = \sqrt{(2r)^2 - r^2} = \sqrt{4r^2 - r^2} = \sqrt{3r^2} = \sqrt{3} \cdot r$$

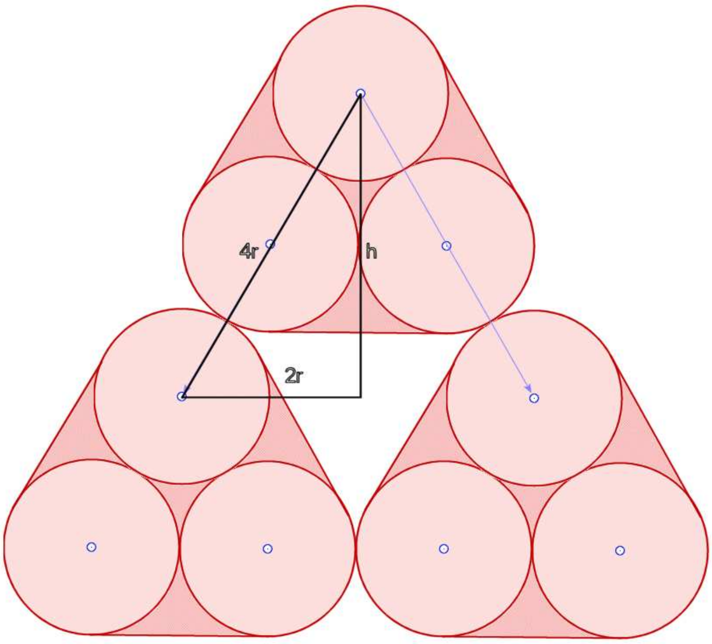
Así pues, partiremos de un punto cualquiera (x_0, y_0) sobre el que construir la iteración

El centro del círculo superior lo situamos sobre dicho punto, por lo que con respecto a este punto no sufre desplazamiento alguno.

El círculo de la izquierda se desplaza h_1 unidades hacia abajo y r unidades hacia la izquierda, es decir, sigue el desplazamiento determinado por el vector $v_i = \left(-r, -\sqrt{3} \cdot r\right)$

El círculo de la derecha se desplaza h_1 unidades hacia abajo y r unidades hacia la derecha, es decir, sigue el desplazamiento determinado por el vector $v_d = \left(+r, -\sqrt{3} \cdot r\right)$

En la segunda iteración,

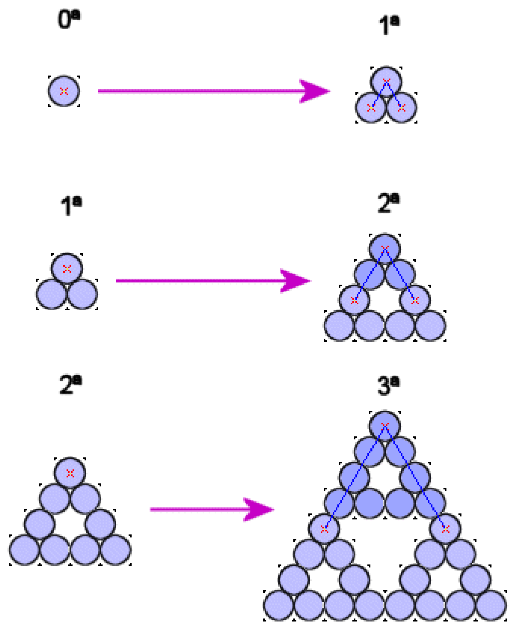


$$h_2 = \sqrt{(4r)^2 - (2r)^2} = \sqrt{16r^2 - 4r^2} = \sqrt{12r^2} = 2\sqrt{3} \cdot r$$

El centro del bloque superior lo situamos sobre el punto (x_0, y_0), por lo que con respecto a este punto no sufre desplazamiento alguno.

El círculo de la izquierda se desplaza h_2 unidades hacia abajo y 2r unidades hacia la izquierda, es decir, sigue el desplazamiento determinado por el vector $v_i = \left(-2r, -2\sqrt{3} \cdot r\right)$

El círculo de la derecha se desplaza h_2 unidades hacia abajo y 2r unidades hacia la derecha, es decir, sigue el desplazamiento determinado por el vector $v_d = \left(+2r, -2\sqrt{3} \cdot r\right)$



En las iteraciones sucesivas, es fácil deducir que:

La hipotenusa de los triángulos de desplazamiento es $a_k = \cdot 2^k \cdot r$

La base del triángulo de desplazamiento es $r_k = 2^{k-1} \cdot r$

La altura del triángulo de desplazamiento es $h_k = 2^{k-1} \cdot \sqrt{3} \cdot r$

El centro del bloque superior lo situamos sobre el punto (x_0, y_0), por lo que con respecto a este punto no sufre desplazamiento alguno.

El círculo de la izquierda se desplaza h_k unidades hacia abajo y r_k unidades hacia la izquierda, es decir, sigue el desplazamiento determinado por el vector $v_i = \left(-r_k, -h_k\right)$

El círculo de la derecha se desplaza h_k unidades hacia abajo y r_k unidades hacia la derecha, es decir, sigue el desplazamiento determinado por el vector $v_d = \left(+r_k, -h_k\right)$

Además, el número de círculos se amplía multiplicando por 3 en número de círculos de la iteración anterior:

$$n_2 = 3 \cdot n_1 = 3 \cdot 3 = 3^2$$

$$n_3 = 3 \cdot n_2 = 3 \cdot 3^2 = 3^3$$

. . .

$n_k = 3^k$

TRATAMIENTO DE LOS BLOQUES POR CADA ITERACIÓN

Cada bloque estará formado por dos matrices de datos, correspondientes a las abscisas y ordenadas de los centros de cada círculo.

Para construir una nueva iteración ampliamos las matrices tres veces de modo que:

- El primer tercio se corresponde con el bloque original

- El segundo tercio, con el bloque desplazado según el vector v_i

- El tercer tercio, con el bloque desplazado según el vector v_d

Tablas de la iteración anterior (k-1):

X_0 de dimensión $n_{k-1} = 3^{k-1}$

Y_0 de dimensión $n_{k-1} = 3^{k-1}$

Tablas de la iteración actual:

desde el índice 1 hasta el 3^{k-1}

$$\begin{cases} X_1 = X_0 \\ Y_1 = Y_0 \end{cases}$$

desde el índice $3^{k-1} + 1$ hasta el $2 \cdot 3^{k-1}$

$$\begin{cases} X_1 = X_0 \\ Y_1 = Y_0 \end{cases} \text{ambos desplazados según } v_i \text{ correspondiente}$$

desde el índice $2 \cdot 3^{k-1} + 1$ hasta el 3^k

$$\begin{cases} X_1 = X_0 \\ Y_1 = Y_0 \end{cases} \text{ambos desplazados según } v_d \text{ correspondiente}$$

Para formar las iteraciones en modo BUCLE, al finalizar cada iteración, seguiremos la secuencia de pasos siguientes:

1º.- Borrar las tablas X_0 e Y_0

2º.- Crear nuevas tablas X_0 e Y_0 vacías

3º.- Copiar las tablas X_1 e Y_1 en X_0 e Y_0 respectivamente

4º.- Borrar las tablas X_1 e Y_1

Para que todo esto funciones correctamente, es necesario establecer el punto de inicio donde las tablas X_0 e Y_0 son de dimensión 1 y contienen las coordenadas del punto cartesiano a partir del cual comenzaremos a crear el triángulo, x_0 e y_0 respectivamente.

Así, si seguimos hasta la iteración 5ª, obtendremos:

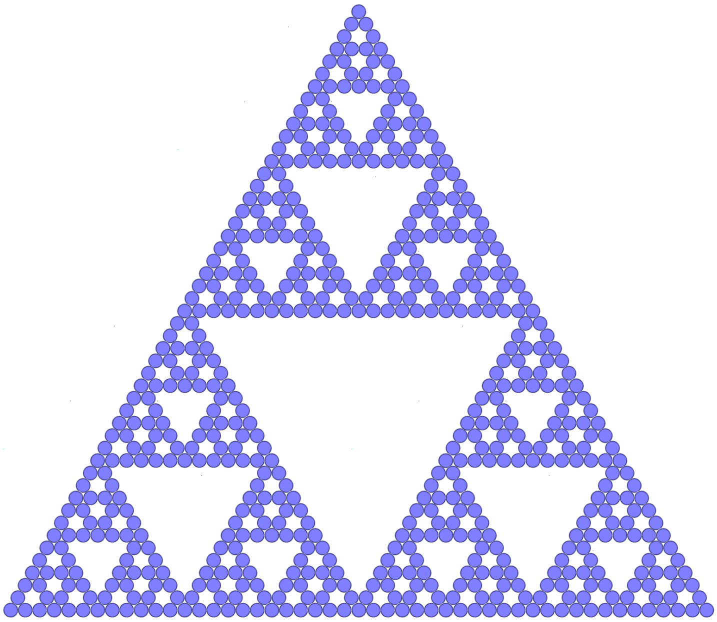
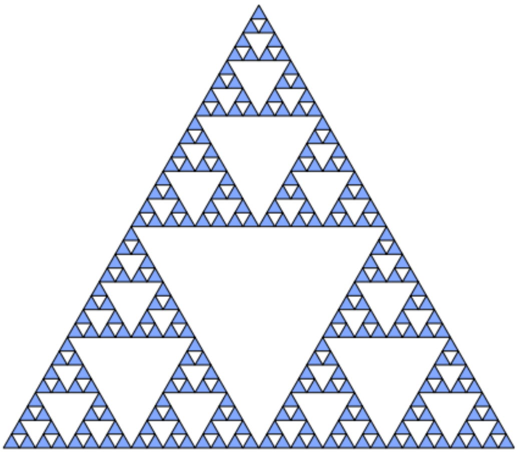
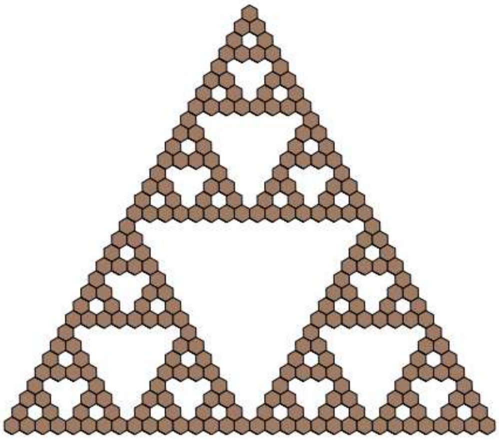


FIGURA BASE

Todo este proceso se ha realizado tomando como figura base un círculo de radio r.

Si en lugar de un círculo, utilizamos un triángulo equilátero o un hexágono regular obtendríamos triángulos de Sierpinski semejantes:





CODIFICACIÓN A PDF

No hay problemas importantes a la hora de codificar en PHP-PDF salvo que el sistema de coordenadas tiene el origen en el vértice superior izquierdo de la hoja; el eje X aumenta hacia la derecha, pero el eje Y funciona al revés, aumenta hacia abajo, por lo que los dibujos saldrán invertidos verticalmente salvo que corrijamos esta diferencia.