

Generación de fragmentos musicales por medio de un algoritmo matemático

Elaborado por:

Ángel Rubén Hernández Quezada

Programa de adscripción:

Ingeniería en sistemas computacionales

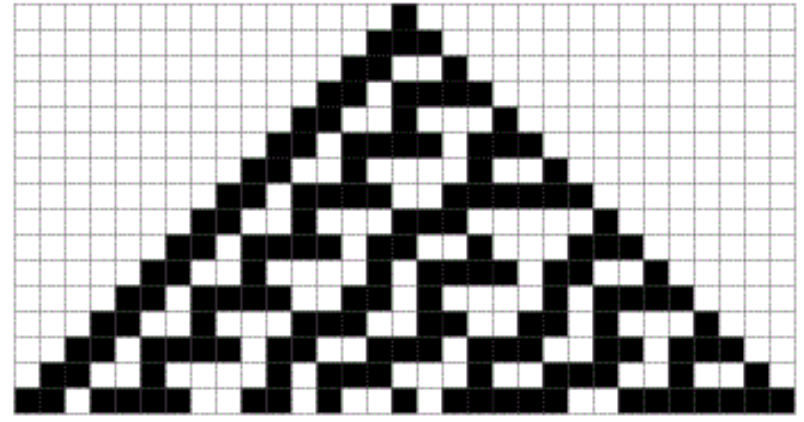
Objetivo:

Desarrollar un prototipo basado en un algoritmo matemático que genere una sucesión de notas musicales



¿Que es un autómata celular?

Básicamente consiste en una sola fila o lista de células a los que se aplica un principio de vecindad básico de dos vecinos por célula y a los que igualmente se pueden aplicar las diversas reglas o condiciones de fronte-

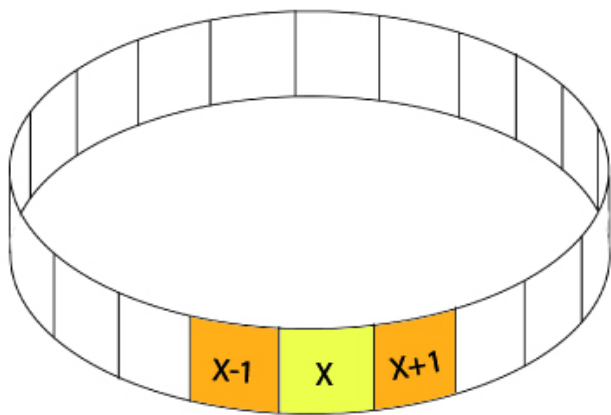


Triángulo de Sierpinski

¿Cómo funciona?

Las reglas que son aplicadas para actualizar las celdas en cada iteración son la clave para que el estilo musical tenga lugar. Estas reglas son las de un autómata celular tradicional basado en el número de celdas vecinas. Musicalmente esto se refiere a la densidad del ritmo si consideramos el siguiente fenómeno: entre más vecinos tenga una semicorchea, más densa es el área del arreglo lineal de celdas en dicha región. Por lo tanto es una buena práctica el variar las reglas para buscar generar patrones más rítmicos que otros.

- ⇒ Si tiene 1 o 2 vecinos permanece igual
- ⇒ Si tiene 3 vecinos y está viva cambia de estado
- ⇒ Si tiene 4 vecinos muere por sobrepoblación.



Representación gráfica de un autómata celular unidimensional de condición de frontera periódica

Además del modelo matemático establecido para ser el motor de composición musical existe el **“mapping”**, este consiste en crear relaciones directas entre el output de un algoritmo (que por lo general genera un output numérico) y parámetros musicales. El mapping de un modo generalizado puede ser concebido como la traducción del algoritmo en música. Este proceso tiene una gran importancia para crear lo que se conoce como un sonido melódico.



Clave de fa

```
public class MiAutomata implements JMC{  
    int longitudarreglo = 10;  
    int[] lon = new int[longitudarreglo];  
    int[] arreglotemporal = new int[arreglotemporal];  
    Part sonido1 = new Part("piano");  
    Score score = new Score();  
}
```

