

Tecnológico de Costa Rica

Proyecto Programado 3
Autómatas Celulares

IC-1801
Taller de Programación
Escuela de Ingeniería en Computación

Prof. Mauricio Avilés

Estudiantes
Axel Fernández Jiménez
2016098894

María José Paz Gómez
2016106862

10/06/2016

Tabla de contenido

MANUAL DE USUARIO	3
PASOS INICIALES	3
DESCRIPCIÓN GENERAL	4
JUEGOS	5
JUEGO DE LA VIDA DE CONWAY	5
Reglas	5
Pasos	6
EL CEREBRO DE BRIAN	7
Reglas	7
Pasos	8
AUTÓMATA CELULAR CÍCLICO	9
Reglas	9
Pasos	10
HORMIGA DE LANGTON	11
Reglas	11
Pasos	12

Manual de Usuario

Se han creado un conjunto de programas para formar diferentes figuras. Este documento es para explicar el funcionamiento de cada uno de ellas, de forma detallada para que se eviten equivocaciones durante el proceso.

Para poder ejecutar los programas de la manera correcta, se deben realizar un conjunto de pasos:

Pasos Iniciales

1. Lo primero que se debe hacer, es descargar Python, ya que es en este lenguaje en el que se programo la información.
2. Para que Python se pueda utilizar, se debe descargar IDLE, con esta aplicación se van a poder correr los programas, de lo contrario no se podrá utilizar el documento de Python brindado.
3. Se debe instalar PyGame, específicamente en Windows.
 - a. Averiguar la versión de Python que se tiene y si es de 32 o 64 bits. Debe ser versión 3.5 o superior.
 - b. Descargar el archivo instalador correspondiente a la versión de Python y ubicarlo en la carpeta conocida.
64 bits: [pygame-1.9.2a0-cp35-none-win_amd64.whl](#)
32 bits: [pygame-1.9.2a0-cp35-none-win32.whl](#)
 - c. Se debe abrir la ventana de la consola.
 - d. Ir a la carpeta Scripts dentro de la instalación de Python (usualmente es C:\Users\<usuario>\AppData\Local\Programs\Python\Python35\Scripts).
 - e. Se debe ejecutar "pip install <dirección completa del archivo que se descargó>", esto instala Pygame en la máquina.
 - f. Probar en el Shell de Python si funciona **"import pygame"**.
4. Para poder usar el documento de Python que contiene los programas, se tiene que guardar en la computadora

El documento de Autómatas Celulares contiene varios programas, cada uno tiene su nombre en rojo y para separarlos se utilizo un conjunto de líneas (-) entre el signo de número (#). Además, cada uno de los programas tiene definido su funcionalidad y también sus entradas, salidas y restricciones para tener una mejor comprensión de cada uno de ellos.

Descripción General

Los autómatas celulares son modelos que consisten en una cuadrícula de células, cada una con un conjunto de estados posibles y células vecinas (vecindario) que determinan su próximo estado. La cuadrícula de células avanza en el tiempo a través de una serie de iteraciones, a partir de un estado inicial predefinido. En cada iteración se determina el estado de cada célula a partir de una función que considera el estado propio y el de sus vecinos. Dependiendo de las reglas utilizadas y el estado inicial las células pueden llegar a formar patrones que persisten a través del tiempo y tienen comportamientos bien definidos.

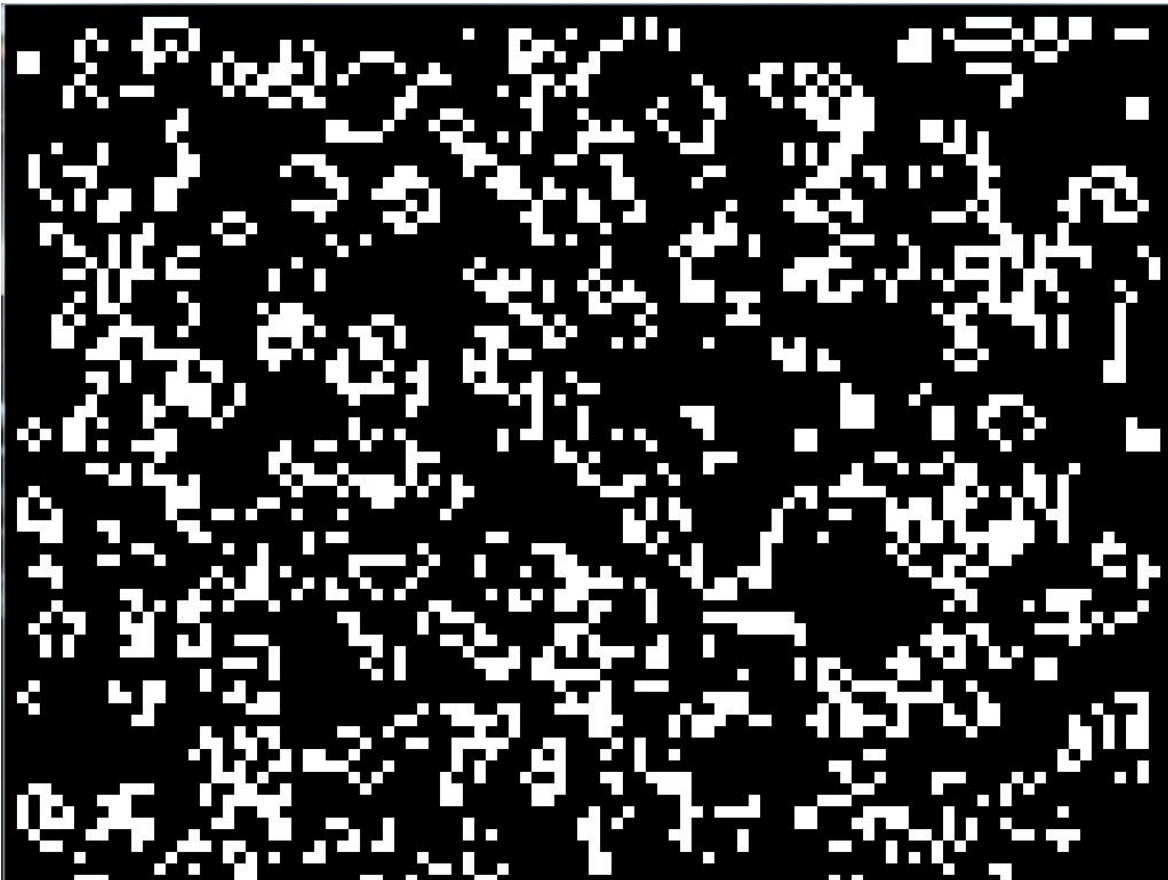
Juegos

Juego de la Vida de Conway

Consiste en una colección de células que, basadas en unas pocas reglas matemáticas, pueden vivir, morir o multiplicarse. Dependiendo de las condiciones originales, las células forman diferentes patrones a través del tiempo. Es una matriz de células, donde cada una puede tener dos posibles estados: viva o muerta. Cada célula interactúa con sus ocho vecinos, que son las celdas horizontal, vertical o diagonalmente adyacentes. El juego evalúa el estado de cada una de las células para determinar su estado en el siguiente paso en el tiempo.

Reglas

- A. **Despoblación:** Una célula viva con menos de dos vecinos vivos se muere
- B. Una célula viva con dos o tres vecinos vivos se mantiene viva con la siguiente generación.
- C. **Sobrepoblación:** Una célula viva con más de tres vecinos vivos muere
- D. **Reproducción:** Una célula muerta con exactamente tres vecinos vivos se convierte en una célula viva



Pasos

1. Se debe poner a correr el programa, esto se hace cuando ya se está en el documento de Fractales y se pulsa la tecla F5, esto hace que aparezca el shell de Python.
2. Para este programa, se debe poner en el primer renglón del shell de Python

```
>>> mainC()
```

3. A continuación se presiona RETURN o ENTER y esto va a hacer que se abra una pestaña que va a mostrar el juego de la Vida de Conway.

Si hubo algún error durante el procedimiento anterior, el programa no va a devolver lo que debe.

```
>>> mainC()  
ERROR
```

El Cerebro de Brian

Este autómata celular es parecido al juego de la vida de Conway, pero modificando las reglas y la cantidad de estados. Al igual que el juego de la vida de Conway, cada célula tiene 8 vecinos. Las células tienen 3 estados: viva, muriendo o muerta.

Reglas

- A. Una célula muerta que tiene exactamente 2 vecinos vivos se convierte en célula viva.
- B. Una célula viva, sin importar la cantidad de vecinos vivos, se convierte en célula que está muriendo.
- C. Una célula que está muriendo, sin importar la cantidad de vecinos vivos, se convierte en célula muerta.



Pasos

1. Se debe poner a correr el programa, esto se hace cuando ya se está en el documento de Fractales y se pulsa la tecla F5, esto hace que aparezca el shell de Python.
2. Para este programa, se debe poner en el primer renglón del shell de Python

```
>>> mainB()
```

3. A continuación se presiona RETURN o ENTER y esto va a hacer que se abra una pestaña que va a mostrar el juego de el Cerebro de Brian.

Si hubo algún error durante el procedimiento anterior, el programa no va a devolver lo que debe.

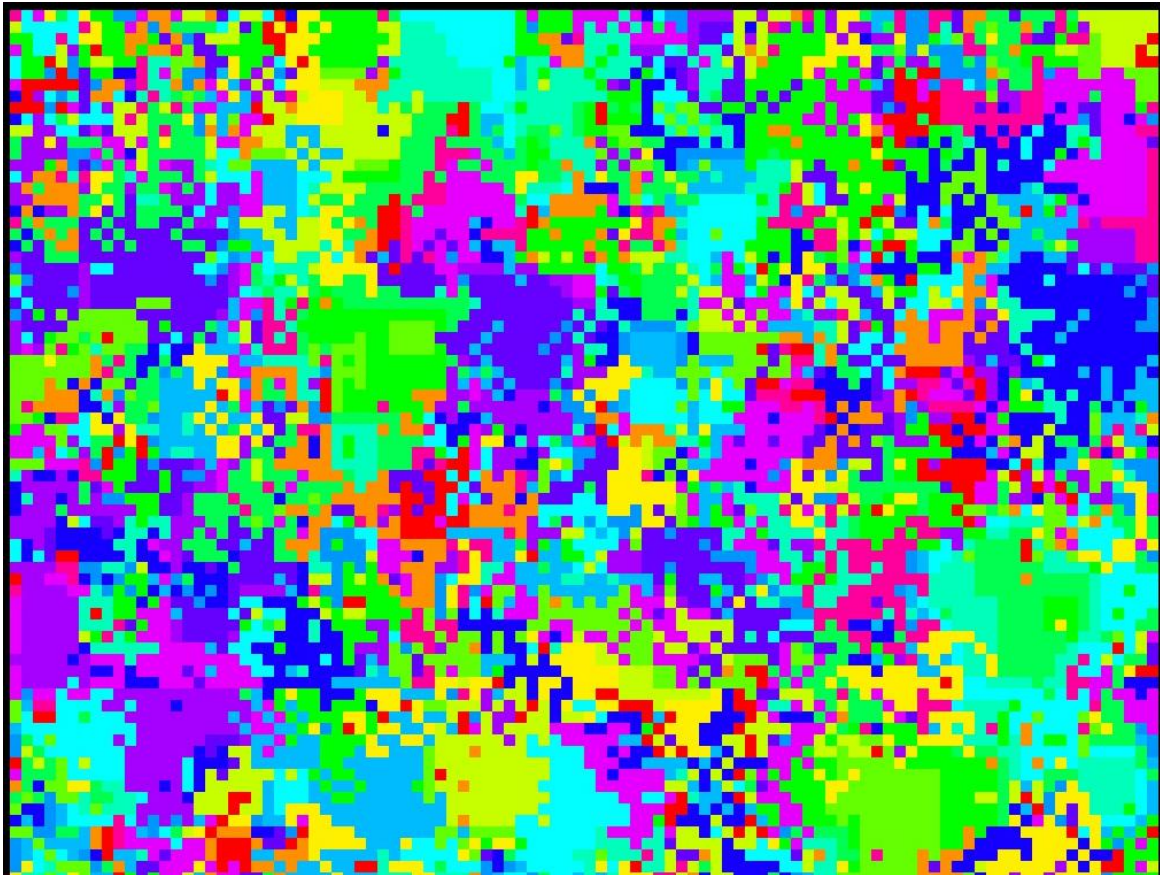
```
>>> mainB()  
ERROR
```


Autómata Celular Cíclico

En este autómata utilizaremos 16 estados para cada célula (de 0 a 15). Cada estado va a estar asociado a un color. Los valores contiguos deben tener colores similares, para dar la sensación de cambio gradual. El siguiente valor después de 15 es 0.

Reglas

- A. Una célula con estado n toma el valor $(n + 1) \% 16$ si tiene al menos un vecino con ese valor.



Pasos

1. Se debe poner a correr el programa, esto se hace cuando ya se está en el documento de Fractales y se pulsa la tecla F5, esto hace que aparezca el shell de Python.
2. Para este programa, se debe poner en el primer renglón del shell de Python

```
>>> mainA()
```

3. A continuación se presiona RETURN o ENTER y esto va a hacer que se abra una pestaña que va a mostrar el juego de Autómata Celular Cíclico.

Si hubo algún error durante el procedimiento anterior, el programa no va a devolver lo que debe.

```
>>> mainA()  
ERROR
```

Hormiga de Langton

Este autómata tiene reglas muy simples, pero genera un comportamiento emergente complejo. Utiliza células con dos estados. En este autómata se maneja una posición dentro de la matriz que representa una hormiga. La hormiga también tiene dirección, puede ir hacia arriba, abajo, izquierda o derecha. Las reglas giran en torno al estado de la célula donde se encuentra la hormiga y la dirección que tiene.

Reglas

- A. Si se encuentra en una célula blanca, gira a la derecha 90 grados, cambia el color de la célula y avanza una célula.
- B. Si se encuentra en una célula negra, gira a la izquierda 90 grados, cambia el color de la célula y avanza una célula.



Pasos

1. Se debe poner a correr el programa, esto se hace cuando ya se está en el documento de Fractales y se pulsa la tecla F5, esto hace que aparezca el shell de Python.
2. Para este programa, se debe poner en el primer renglón del shell de Python

```
>>> mainL()
```

3. A continuación se presiona RETURN o ENTER y esto va a hacer que se abra una pestaña que va a mostrar el juego de La Hormiga de Langton.

Si hubo algún error durante el procedimiento anterior, el programa no va a devolver lo que debe.

```
>>> mainL()  
ERROR
```