

Proyecto 0: el juego de la vida de Conway

Banco de Guatemala

Maestría en Economía y Finanzas Aplicadas

Pogramación I

Profesor: Rodrigo Chang

Profesor auxiliar: Mauricio Vargas

Fecha: Enero 2020

El siguiente proyecto tiene como finalidad de que ejercite los siguientes puntos:

1. Asignación de variables
2. Manejo de diferentes tipos de objetos
3. Condiciones
4. Loops
5. Lectura e impresión de archivos
6. Impresión de resultados en consola

¿Qué es el juego de la vida de Conway?

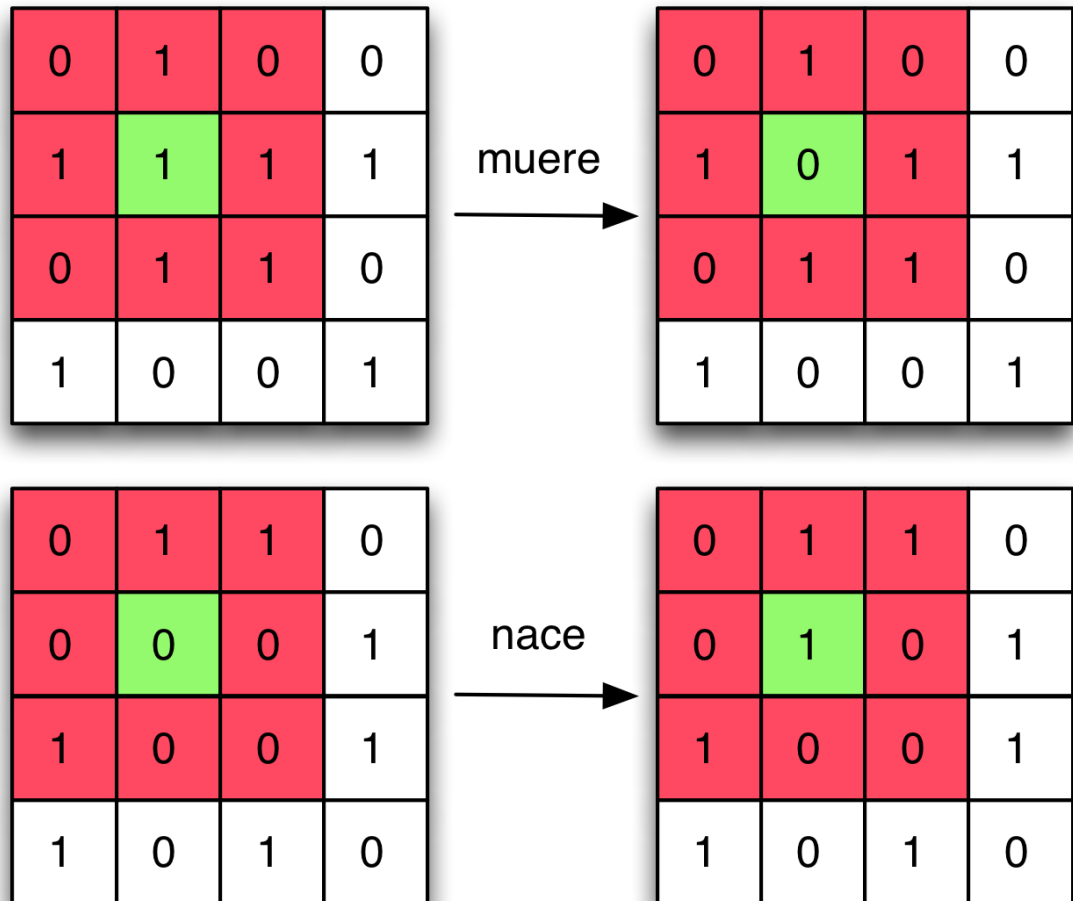
El siguiente vídeo da las intuiciones generales

https://www.youtube.com/watch?v=OWXD_wJxCKQ (https://www.youtube.com/watch?v=OWXD_wJxCKQ)

Se trata de un juego de cero jugadores, lo que quiere decir que su evolución está determinada por el estado inicial y no necesita ninguna entrada de datos posterior. El "tablero de juego" es una malla plana formada por cuadrados (las "células") que se extiende por el infinito en todas las direcciones. Por tanto, cada célula tiene 8 células "vecinas", que son las que están próximas a ella, incluidas las diagonales. Las células tienen dos estados: están "vivas" o "muertas" (o "encendidas" y "apagadas"). El estado de las células evoluciona a lo largo de unidades de tiempo discretas (se podría decir que estos corresponden a los turnos). El estado de todas las células se tiene en cuenta para calcular el estado de las mismas al turno siguiente. Todas las células se actualizan simultáneamente en cada turno, siguiendo estas reglas:

- **Una célula muerta con exactamente 3 células vecinas vivas "nace" (es decir, al turno siguiente estará viva).**
- **Una célula viva con 2 o 3 células vecinas vivas sigue viva, en otro caso muere (por "soledad" o "superpoblación").**

El Juego de la Vida (John Conway, 1970)



Ejemplo de implementación de reglas en el Juego de la Vida

Primera parte: el juego de la vida ¶

Se solicita que programe el *Juego de la Vida* en *Python* siguiendo los siguientes lineamientos:

- Se deben introducir las condiciones iniciales por medio de un archivo de texto. (Adjunto un archivo de prueba que se utilizará como estándar.
 - Dimensión en x y y de la cuadrícula total (matriz).
 - Cantidad de periodos T a jugar.
 - Las siguientes líneas corresponden a las coordenadas (x, y) de las casillas vivas en el tiempo cero ($t_0 = 0$)
- El programa debe correr en consola, solicitando la dirección del archivo de texto con las condiciones iniciales. Por ejemplo, para llamar al programa debe ejecutarse en la terminal:

```
>python.exe nombre_apellido.py
```

- El programa debe generar un archivo tipo `.csv` que contendrá la cantidad total de casillas vivas para cada tiempo t . La primera fila debe contener el encabezado.
 - El nombre del archivo debe seguir el siguiente formato: *nombre_apellido.csv*

Tiempo	Población
0	P_0
\vdots	\vdots
T	P_T

- Finalmente, el programa debe imprimir una matrix en donde, si la casilla está viva, se coloca un `*`, de lo contrario se deja un espacio en blanco. Esto debe realizarse por cada periodo jugado (de 0 a T). Dichas impresiones deben realizarse de un modo secuencial siguiendo los siguientes pasos:
 1. Imprimir la matrix
 2. Limpiar la vista en la consola
 3. Crear un tiempo de espera (*delay*)
 4. Imprime la siguiente matrix

Esto con el fin de dar el efecto de animación en consola.

Debe entregar un script `.py` nombrado con el siguiente formato: `nombre_apellido.py`

Rúbrica de evaluación

Aspecto a evaluar	Punteo
Archivo <code>.csv</code>	10
Lectura de archivo de texto	10
Impresión de matrix	10
Animación	10
Correcta implementación de reglas	60
Total	100

Segunda parte: modelo de influencia (30 puntos extra)

En un salón de clase se discute el resultado de la contienda entre Ernest “Iron” Bran y Esduin “3Kiebres” Javier. La opinión está dividida y todos los alumnos discuten entre sí para convencer a sus compañeros que su elección de púgil victorioso es la mejor. El catedrático nota que, si alrededor de un estudiante se sientan más de cuatro compañeros con preferencias contrarias, el estudiante cambia su preferencia.

En el salón hay 100 alumnos sentados en 10 filas. El periodo de clase tiene una duración de 50 minutos y el catedrático asume que un alumno es capaz de cambiar de preferencia cada 5 minutos de discusión.

Debe modelar el salón de clase como un juego de la vida. La rúbrica es la misma y debe generar un archivo de salida .csv nombrado como *nombre_apellido_bran3k.csv*.

Para entregarlo tiene las siguientes opciones:

1. Se solicita que adapte el programa original para recibir argumentos de la línea de comandos utilizando el paquete `sys`. Esta funcionalidad se debe ejecutar con la opción `--modelo-influencia` al ejecutar el programa. Por ejemplo:

```
>python.exe nombre_apellido.py --modelo-influencia
```

2. Crear un nuevo *script* .py nombrado con el siguiente formato: `nombre_apellido_2.py`