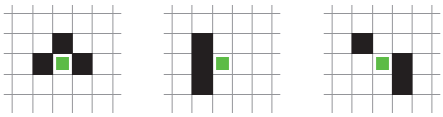
Objetivos

1. Familiarizarse con las matrices como estructuras de datos de dos dimensiones.
2. Ejercitar la implementación de algoritmos de recorrido de matrices.
3. Fomentar la habilidad de descomponer un problema en subproblemas y de implementar funciones que los resuelven, lo que se conoce comúnmente como la técnica de "Dividir y Conquistar".

Contexto: el juego de la vida

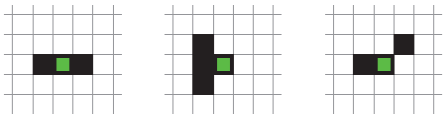
El juego de la vida es un juego sin jugadores. Se trata de colocar una serie de fichas en un tablero y dejar que evolucionen siguiendo unas reglas muy simples. En el juego original se utiliza un tablero (una matriz) con infinitas filas y columnas. Como disponer de una matriz de dimensión infinita en un programa es imposible, supondremos que presenta dimensión m × n, donde m y n son valores dados en un archivo que contiene la información del tablero de juego. Cada casilla del tablero contiene una célula que puede estar viva o muerta. En las siguientes figuras se representan las células vivas con casillas de color negro y las células muertas con casillas en blanco. Cada casilla del tablero cuenta con ocho celdas vecinas. El mundo del juego de la vida está gobernado por una serie de jugadas en las que mueren y nacen células. Cuándo nace y cuándo muere una célula solo depende de cuántas células vecinas están vivas. Estas son las reglas:

1. Regla de nacimiento. Una célula muerta resucita si tiene exactamente tres células vecinas vivas, de lo contrario continúa muerta. En la siguiente figura se muestran (en verde) celdas que están muertas y que pasan a estar vivas en la siguiente jugada:

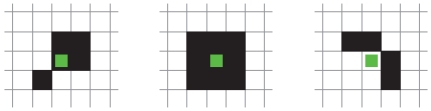


2. Regla de supervivencia. Una celda viva permanece viva solamente si tiene dos o tres células vecinas vivas, de lo contrario muere.

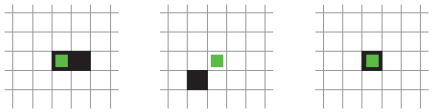
   En la siguiente figura se muestran (en verde) células que están vivas y así permanecerán tras la siguiente jugada:



   En la siguiente figura se muestran (en verde) células que están vivas o muertas y estarán muertas tras la siguiente jugada:



   En la siguiente figura se muestran (en verde) células que están vivas o muertas y estarán muertas tras la siguiente jugada:



Su reto es hacer un programa que muestre la evolución del juego de la vida durante una serie de jugadas.

Para esto:

- El tablero de juego debe ser cargado de un archivo.
- Las células se guardarán en una matriz de valores enteros. Una casilla con valor 1 representará una célula viva y una casilla con valor 0 representará una célula muerta.
- Para dibujar el tablero de juego en pantalla, usaremos un punto para representar una célula muerta y un asterisco para representar una célula viva.
- El usuario determina cuándo desea terminar el juego y cuándo desea realizar una jugada (a través del menú de opciones de la aplicación).
- El juego de la vida original asume que el tablero es infinito. En este caso debemos jugar con un tablero que tiene límites, así que hay que tratar de modo especial las casillas de los bordes del tablero (matriz).
- El tablero de juego NO se puede modificar durante la aplicación de las reglas. Esto quiere decir que se necesitan dos tableros (matrices). Las reglas se calculan sobre el tablero de juego original (es decir, que en este se cuentan las células vecinas vivas) y el resultado de aplicar las reglas se guarda en un segundo tablero. Cuando finaliza el proceso, el segundo tablero se asigna al tablero de juego.

Preparación

Cree una carpeta de trabajo y descargue allí el archivo **`n4-l1-esqueleto.zip`** que se encuentra adjunto a este enunciado en sicuaplus. Descomprima este archivo y abra desde Spyder los archivos **`librería_juego.py`** e **`interfaz_consola.py`**.

Instrucciones

Complete las funciones que se encuentran declaradas en el módulo **librería_juego.py** teniendo en cuenta la documentación y la información que viene a continuación. En este módulo encontrará dos funciones ya implementadas: cargar_tablero y regla_supervivencia. Usted debe completar las otras tres funciones: cuantas_vecinas_vivas, regla_nacimiento y realizar_jugada.

Note que se le está entregando la interfaz basada en consola (**interfaz_consola.py**) completamente implementada. Abra el archivo y familiarícese con el código de las diferentes funciones: `iniciar_aplicacion`, `mostrar_menu`, `ejecutar_cargar_tablero`, `pintar_tablero` y `ejecutar_jugada`.

Puede probar la implementación de sus funciones cargando la información que se encuentra en los archivos "**tablero1.csv**", "**tablero2.csv**" y "**tablero3.csv**".

Problema 1: Contar las células vecinas vivas

Complete la función `cuantas_vecinas_vivas` que recibe como parámetro un tablero de juego y las coordenadas (fila y columna) de una casilla del tablero y retorna la cantidad de células vecinas vivas.

ATENCIÓN: recuerde que al explorar las casillas vecinas de una casilla dada, usted debe verificar que no se salga de los límites de la matriz.

Problema 2: Verificar regla de nacimiento

Complete la función `regla_nacimiento` que recibe como parámetro un tablero de juego y las coordenadas (fila y columna) de una casilla del tablero y retorna verdadero si la célula que se encuentra en dicha casilla cumple con la regla de nacimiento. Esto es: si la célula en la casilla está muerta y debe resucitar según la regla de nacimiento, el valor de retorno es True, de lo contrario es False.

Problema 3: Verificar regla de supervivencia

Estudie la función `regla_supervivencia` que recibe como parámetro un tablero de juego y las coordenadas (fila y columna) de una casilla del tablero y retorna verdadero si la célula que se encuentra en dicha casilla cumple con la regla de supervivencia. Esto es: si la célula en la casilla está viva y debe permanecer viva, el valor de retorno es True. Si la célula debe morir, el valor de retorno es False.

Problema 4: Realizar una jugada

Complete la función `realizar_jugada` que recibe como parámetro un tablero de juego, simula una jugada (aplicando las reglas del juego de la vida sobre cada casilla del tablero) y retorna el tablero de juego modificado tras la jugada. Esta función debe utilizar OBLIGATORIAMENTE las funciones desarrolladas anteriormente para contar las células vecinas de cada casilla del tablero y aplicar las reglas del juego.

Pruebe el correcto funcionamiento de su programa

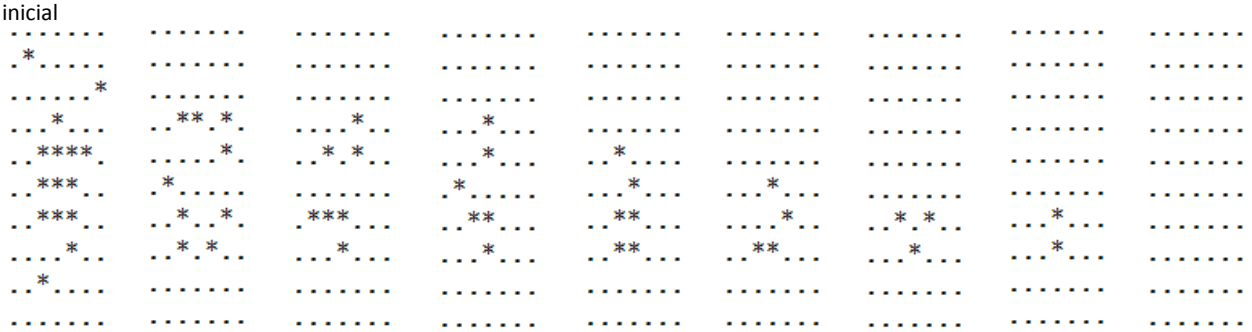
A continuación se muestra la evolución del tablero de juego, iniciando el juego con cada uno de los 3 tableros de prueba para que pueda comprobar visualmente si el comportamiento de su juego es correcto o no.

Evolución del juego con tablero1.csv

Tablero inicial

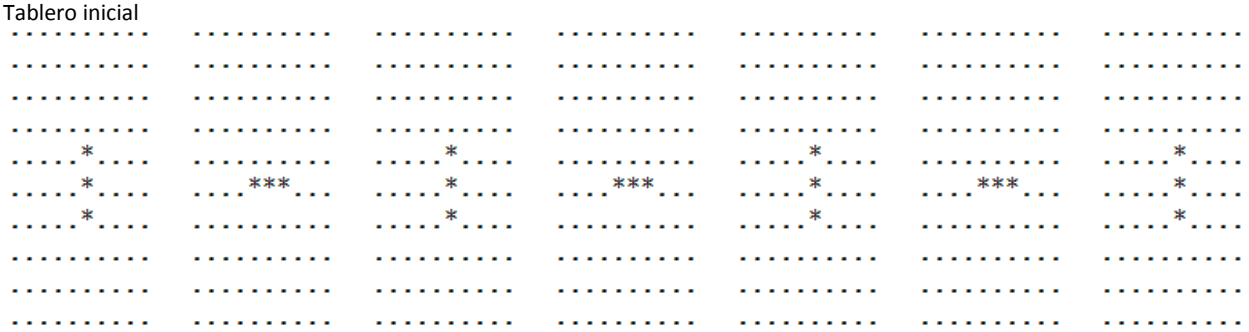
Evolución del juego con tablero2.csv

Tablero inicial



Evolución del juego con tablero2.csv

Tablero inicial



Entrega

Entregue el archivo modificado a través de Sicuaplus en la tarea designada como “**N4-L1**”.