2025-1

Tarea 4

EL JUEGO DE LA VIDA

Estructura de Datos
Universidad Nacional Autónoma de México
Facultad de Estudios Superiores Aragón

REALIZADO POR:

Enrique Emiliano Cano García

22/08/2024

Clase Array2D.

```python
class Array2D:  3 usages  kno4
    def __init__(self, ren, col):  kno4
        self._data = [[None] * col for _ in range(ren)]
        self._ren = ren
        self._cols = col

    def clear(self, dato):  2 usages  kno4
        for row in range(self._ren):
            for col in range(self._cols):
                self._data[row][col] = dato

    def get_ren_size(self):  kno4
        return self._ren

    def get_col_size(self):  kno4
        return self._cols

    def set_item(self, ren, col, dato):  5 usages  kno4
        if 0 <= ren < self._ren and 0 <= col < self._cols:
            self._data[ren][col] = dato
        else:
            raise IndexError("Índice fuera de rango")


    def get_item(self, ren, col):  2 usages  kno4
        if 0 <= ren < self._ren and 0 <= col < self._cols:
            return self._data[ren][col]
        else:
            raise IndexError("Índice fuera de rango")

    def to_string(self):  1 usage  kno4
        result = ""
        for row in self._data:
            result += " ".join(map(str, row)) + "\n"
        return result

```

Juego de la vida

```python
from Array2D import Array2D

class JuegoDeLaVida:  1 usage  kno4
    def __init__(self, rens, cols):  kno4
        self.rejilla = Array2D(rens, cols)
        self.rejilla.clear(0)
        self.rens = rens
        self.cols = cols

    def configurar(self, lista_celulas_vivas):  1 usage  kno4
        """
        lista_celulas_vivas debe ser una lista de tuplas [(ren, col), (ren, col), ...]
        que especifican las posiciones de las células vivas.
        """
        for ren, col in lista_celulas_vivas:
            self.rejilla.set_item(ren, col, 1)  # Establece la célula como viva (1)

    def get_num_vecinos_vivos(self, ren, col):  1 usage  kno4
        vecinos_vivos = 0
        for i in range(ren - 1, ren + 2):
            for j in range(col - 1, col + 2):
                if (i == ren and j == col) or i < 0 or j < 0 or i >= self.rens or j >= self.cols:
                    continue
                if self.rejilla.get_item(i, j) == 1:
                    vecinos_vivos += 1
        return vecinos_vivos
    def actualizar_generacion(self):  1 usage  kno4
        nueva_grid = Array2D(self.rens, self.cols)
        nueva_grid.clear(0)  # Inicializamos la nueva rejilla con todas las células muertas

        for ren in range(self.rens):
            for col in range(self.cols):
                vecinos_vivos = self.get_num_vecinos_vivos(ren, col)
                estado_actual = self.rejilla.get_item(ren, col)

                # Aplicación de las reglas del juego
                if estado_actual == 1:  # Célula viva
                    if vecinos_vivos == 2 or vecinos_vivos == 3:
                        nueva_grid.set_item(ren, col, 1)  # Sobrevive
                    else:
                        nueva_grid.set_item(ren, col, 0)  # Muere
                else:  # Célula muerta
                    if vecinos_vivos == 3:
                        nueva_grid.set_item(ren, col, 1)  # Nace una nueva célula
                    else:
                        nueva_grid.set_item(ren, col, 0)  # Permanece muerta

        # Actualizamos la rejilla con la nueva generación
        self.rejilla = nueva_grid

    def jugar(self, generaciones):  1 usage  kno4
        """
        Juega el juego durante un número de generaciones.
        """
        for gen in range(generaciones):
            print(f"Generación {gen + 1}:")
            print(self.rejilla.to_string())  # Imprime el estado actual de la rejilla
            self.actualizar_generacion()
```

Main

```python
if __name__ == "__main__":
    # Inicializar el juego con una rejilla de 5x5
    juego = JuegoDeLaVida(10, 10)

    celulas_vivas = [(2, 1), (2, 2), (2, 3), (3,4), (8,4), (7,4), (3,5)]
    juego.configurar(celulas_vivas)

    # Jugar durante 5 generaciones
    juego.jugar(10)
```

Ejecución

```
Generación 1:
0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0
0 1 1 1 0 0 0 0 0 0
0 0 0 0 1 1 0 0 0 0
0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0
0 0 0 0 1 0 0 0 0 0
0 0 0 0 1 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0

Generación 2:
0 0 0 0 0 0 0 0 0 0
0 0 1 0 0 0 0 0 0 0
0 0 1 1 1 0 0 0 0 0
0 0 1 1 1 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0
```

```
Generación 3:
0 0 0 0 0 0 0 0 0 0
0 0 1 0 0 0 0 0 0 0
0 1 0 0 1 0 0 0 0 0
0 0 1 0 1 0 0 0 0 0
0 0 0 1 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0

Generación 4:
0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0
0 1 1 0 0 0 0 0 0 0
0 0 1 0 1 0 0 0 0 0
0 0 0 1 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0
```

```
Generación 5:
0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0
0 1 1 1 0 0 0 0 0 0
0 1 1 0 0 0 0 0 0 0
0 0 0 1 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0

Generación 6:
0 0 0 0 0 0 0 0 0 0
0 0 1 0 0 0 0 0 0 0
0 1 0 1 0 0 0 0 0 0
0 1 0 0 0 0 0 0 0 0
0 0 1 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0
```

```
Generación 7:
0 0 0 0 0 0 0 0 0 0
0 0 1 0 0 0 0 0 0 0
0 1 0 0 0 0 0 0 0 0
0 1 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0

Generación 8:
0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0
0 1 1 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0
```

```
Generación 9:
0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0

Generación 10:
0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0
```