



CENTRO DE CIENCIAS BÁSICAS
DEPARTAMENTO DE CIENCIAS DE LA COMPUTACIÓN
AUTÓMATAS II
7° "A"

Regla de la Mayoría

Dr. Francisco Javier Ornelas Zapata

Alumnos:

Almeida Ortega Andrea Melissa
Espinoza Sánchez Joel Alejandro
Flores Fernández Óscar Alonso
Gómez Garza Dariana
González Arenas Fernando Francisco
Orocio García Hiram Efraín

Fecha de Entrega: Aguascalientes, Ags., **26** de noviembre de 2021

Se genera un programa que utilice un autómata celular (AC) para ejemplificar la regla de la mayoría.

La tabla de este autómata celular es la siguiente:

000	001	010	011	100	101	110	111
0	0	0	1	0	1	1	1

Efectivamente, cada autómata calcula cuál es el estado mayoritario en su entorno y se actualiza de acuerdo con la mayoría. En directa la interpretación de los estados 0 y 1 como dos opiniones diferentes en una población o inclinaciones políticas, de modo que el autómata celular se convierte en modelo de evolución de opiniones o de intención de voto.

Obviamente, en este modelo no tiene mucho sentido partir de la semilla inicial considerada en los dos casos anteriores. Es mucho más interesante ver cómo evoluciona a partir de una condición inicial al azar y ver si alguno de los dos estados termina imponiéndose al otro.

Otra forma de como se conoce esto es como “Pánico”, ya que es otra forma de ver o de explicar esto ya que en este caso cada célula representa un persona. La persona tiene dos estados ya que como sabemos esto es un modelo binario, o esta “tranquilo (0)” o en pánico (1).

Reglas:

1. Cuando haya 4 o más vecinos en pánico se hará pasara de estado tranquilo a en pánico.
2. Cuando haya al menos 3 vecinos activos (en pánico), si hay menos se tranquiliza.
3. La celda central cambia según la mayoría de sus vecinos.

Se va a presentar un código en Python donde se podrá representar esto.

```
def convert(d):
```

```
    if (d == "000"):
```

```
        { 2 }
```

```

        r = '0';
    elif (d == "001"):
        r = '0';
    elif (d == "010"):
        r = '0';
    elif (d == "011"):
        r = '1';
    elif (d == "100"):
        r = '0';
    elif (d == "101"):
        r = '1';
    elif (d == "110"):
        r = '1';
    elif (d == "111"):
        r = '1';
    return r;

# print convert("001")
#cad = "101011100101000"

cad = input("Cadena de entrada (0 y 1): ")
ran = input("Cantidad de iteraciones:")
t = len(cad)
index = t - 1
print("La longitud de tu cadena es: ", t)
r = ""
for j in range(int(ran)):
    aux1 = index # este debe ser longitud -1 , al iniciar
    aux2 = index - index # este debe ser 0 al iniciar
    aux3 = index - (index - 1) # este debe ser 1 al iniciar
    cnt = 1
    r = ""
    for i in range(t):
        if aux3 == t:
            aux3 = 0
            l = cad[aux1]

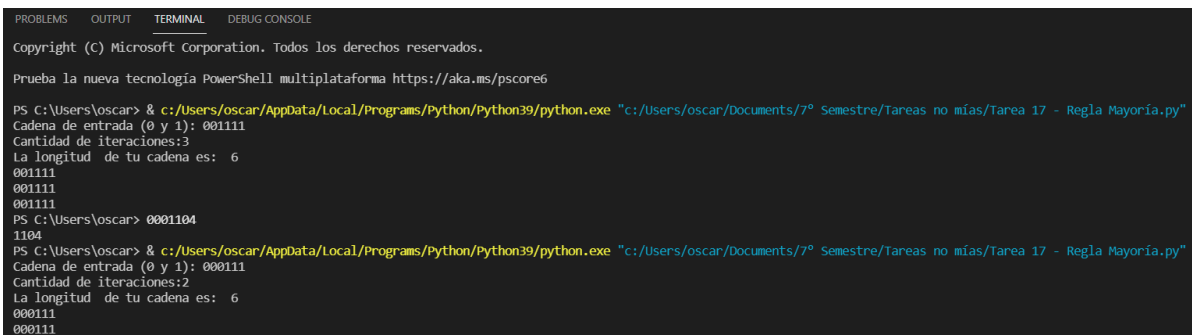
```

```

m = cad[aux2]
n = cad[aux3]

n1 = 1 + m + n
if cnt == 1:
    aux1 = 0
    aux2 = 1
    aux3 = 2
else:
    aux1 = aux1 + 1
    aux2 = aux2 + 1
    aux3 = aux3 + 1
cnt = cnt + 1
r += convert(n1)
cad = r
print(r)

```



```

PROBLEMS  OUTPUT  TERMINAL  DEBUG CONSOLE
Copyright (C) Microsoft Corporation. Todos los derechos reservados.
Prueba la nueva tecnología PowerShell multiplataforma https://aka.ms/pscore6

PS C:\Users\oscar> & c:/Users/oscar/AppData/Local/Programs/Python/Python39/python.exe "c:/Users/oscar/Documents/7º Semestre/Tareas no mías/Tarea 17 - Regla Mayoría.py"
Cadena de entrada (0 y 1): 001111
Cantidad de iteraciones:3
La longitud de tu cadena es: 6
001111
001111
001111
PS C:\Users\oscar> 0001104
1104
PS C:\Users\oscar> & c:/Users/oscar/AppData/Local/Programs/Python/Python39/python.exe "c:/Users/oscar/Documents/7º Semestre/Tareas no mías/Tarea 17 - Regla Mayoría.py"
Cadena de entrada (0 y 1): 000111
Cantidad de iteraciones:2
La longitud de tu cadena es: 6
000111
000111

```