

Symulator rzutu kostką

Łukasz Kopka, Mateusz Kowalski

Zasada działania programu

1. Wylosowanie liczby całkowitej z przedziału $\langle 1, 6 \rangle$
2. Wyświetlenie wylosowanej liczby wraz z graficzną prezentacją wylosowanej ścianki kostki

Biblioteki i podstawowe zmienne

```
import random //importowanie biblioteki random, pozwalającej na użycie  
               m.im. funkcji random.randint()
```

```
import os //importowanie biblioteki os, pozwalającej na użycie  
          m.im. funkcji os.system('CLS')
```

```
stars = "*" * 10 //deklaracja zmiennej służącej jako linia oddzielająca elementy
```

```
border = "X" * 9 //deklaracja zmiennej służącej jako górne i dolne obramowanie  
                  graficznej wizualizacji kostki
```

Losowanie liczby

```
oczka = random.randint(1, 6) //nadanie zmiennej „oczka” wartości  
                               wylosowanej liczby
```

```
print("Wylosowano: ")
```

```
print(oczka)
```

```
if oczka == 1:
```

```
    row1 = "X" + "    " + "X "
```

```
    row2 = "X" + " 0  " + "X "
```

```
    row3 = "X" + "    " + "X "
```

```
//sprawdzenie wartości wylosowanej  
liczby i ustawienie dla zmiennych  
row1, row2, row3 ciągu znaków  
służących za graficzną prezentację kostki
```

elif oczka == 2:

row1 = "X" + " O " + "X"

row2 = "X" + " " + "X"

row3 = "X" + " O " + "X"

elif oczka == 3:

row1 = "X" + " O " + "X"

row2 = "X" + " O " + "X"

row3 = "X" + " O " + "X"

elif oczka == 4:

row1 = "X" + " O O " + "X"

row2 = "X" + " " + "X"

row3 = "X" + " O O " + "X"

elif oczka == 5:

row1 = "X" + " O O " + "X"

row2 = "X" + " O " + "X"

row3 = "X" + " O O " + "X"

else:

row1 = "X" + " O O " + "X"

row2 = "X" + " O O " + "X"

row3 = "X" + " O O " + "X"

Wyświetlenie wyniku

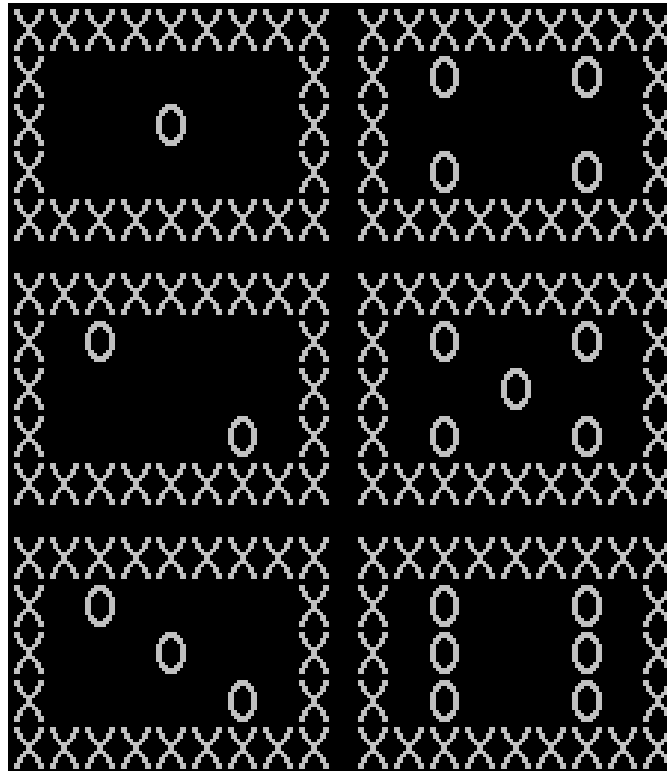
```
print(border)
```

```
print(row1)
```

```
print(row2)
```

```
print(row3)
```

```
print(border)
```



Kod

Cały kod dostępny jest pod adresem:

<https://github.com/Matkov569/kostka/blob/master/kostka.py>

The background features abstract, overlapping green geometric shapes, primarily triangles and polygons, in various shades of green, creating a modern and dynamic visual effect.

Dziękujemy za uwagę

Łukasz Kopka, Mateusz Kowalski