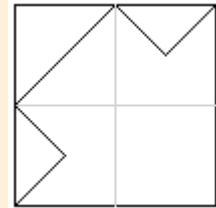


Przedmiotowy Konkurs Informatyczny LOGIA powołany przez Mazowieckiego Kuratora Oświaty

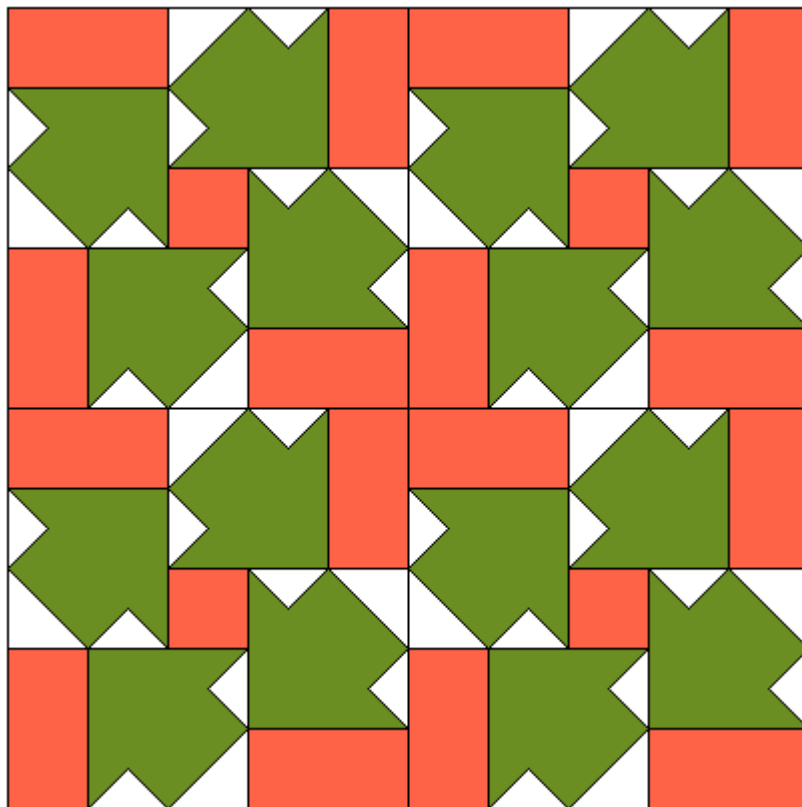
Zadanie Dywan – LOGIA 22 (2021/22), etap 1

Treść zadania

Napisz funkcję **dywan()**, po wywołaniu której powstanie na środku ekranu rysunek dywanu taki, jak poniżej. Długość boku dywanu wynosi **400**. Proporcje poszczególnych elementów odczytaj z rysunku pomocniczego.



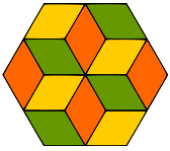
rysunek pomocniczy



Efekt wywołania: **dywan()**

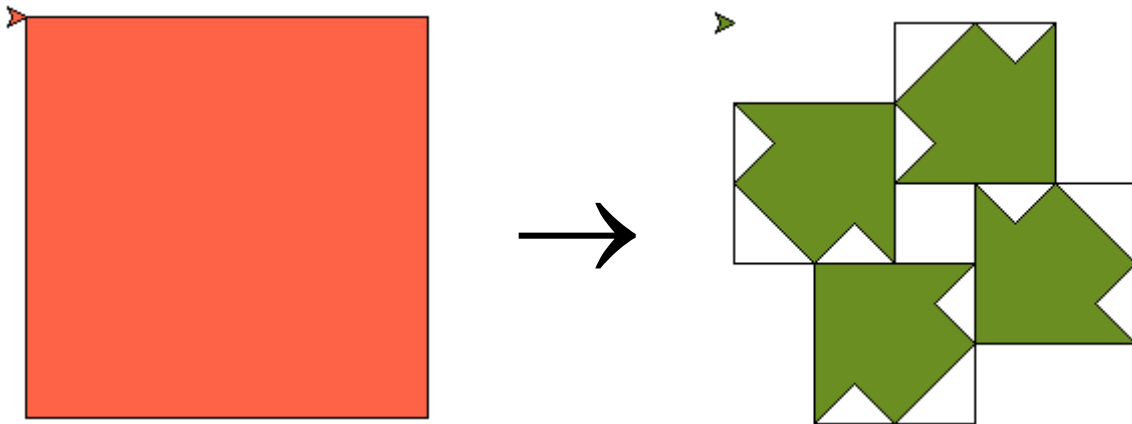
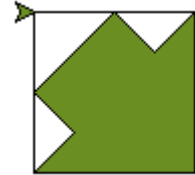
Omówienie rozwiązania

Zadanie polega na narysowaniu dywanu złożonego z czterech identycznych elementów. Aby sprostać wymogom ustawienia rysunku na środku ekranu warto zaczynać rysowanie jednego powtarzającego się elementu zaczynając od takiego rogu, aby kolejne elementy były rysowane po obrocie o 90° .

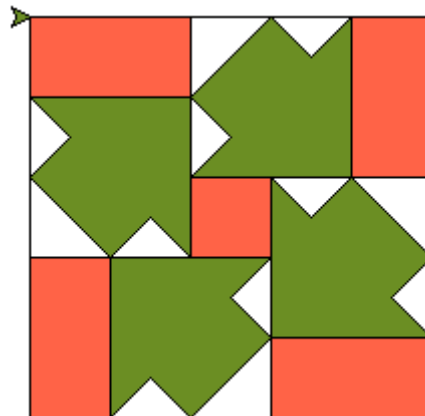


Przedmiotowy Konkurs Informatyczny LOGIA powołany przez Mazowieckiego Kuratora Oświaty

Można zacząć rysowanie powtarzanego elementu od narysowania zamalowanego kolorem czerwonym kwadratu o boku 200 (połowa rozmiaru). Następnie na tym kwadracie narysować układ czterech strzałek – każda w zamalowanym kolorem białym kwadracie o boku 80. Jako funkcje pomocnicze warto zdefiniować funkcję rysowania zamalowanego kwadratu, gdzie parametrami będą bok kwadratu i kolor zamalowania oraz funkcję, która narysuje zieloną strzałkę w białym kwadracie.



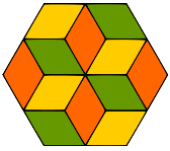
Rysowanie jednego elementu dywanu – funkcja **jeden()**.



Efekt wywołania funkcji **jeden()**

Rozwiązanie w języku Python

```
1 from turtle import *
2 from math import sqrt
3
4 def kwad(bok, kolor):
5     fillcolor(kolor)
6     begin_fill()
7     for i in range(4):
8         fd(bok); rt(90)
9     end_fill()
10
```



Przedmiotowy Konkurs Informatyczny LOGIA powołany przez Mazowieckiego Kuratora Oświaty

```
11 def strz():
12     bok = 80
13     kwad(bok, 'white')
14     fd(bok/2)
15     fillcolor('olivedrab')
16     rt(45)
17     begin_fill()
18     fd(bok * sqrt(2) / 4); lt(90)
19     fd(bok * sqrt(2) / 4); rt(135)
20     fd(bok); rt(90); fd(bok); rt(135)
21     fd(bok * sqrt(2) / 4); lt(90)
22     fd(bok * sqrt(2) / 4); rt(90)
23     fd(bok * sqrt(2) / 2); rt(45); bk(bok/2)
24     end_fill()
25
26 def jeden():
27     kwad(200, 'tomato')
28     pu(); fd(80); pd()
29     for i in range(4):
30         strz()
31         pu(); fd(120); rt(90); fd(80); pd()
32     pu(); bk(80); pd()
33
34 def dywan():
35     for i in range(4):
36         jeden()
37         rt(90)
```

Testy

Ponieważ funkcja **dywan()** jest bezparametrowa, wystarczy raz przetestować działanie funkcji. Sprawdzamy prawidłowość rysunku, jego zgodność w treścią zadania, rozmiar i wyśrodkowanie.

W języku Python, aby przyspieszyć tworzenie rysunku przez żółwia, stosujemy wywołanie złożone z funkcji **tracer()** – rysownie w pamięci, właściwego wywołania funkcji **dywan()** i na końcu uaktualniamy ekran za pomocą funkcji **update()**. Przykład:

```
tracer(0)
dywan()
update()
```