

\*\*\*\*\*

# Tema 4

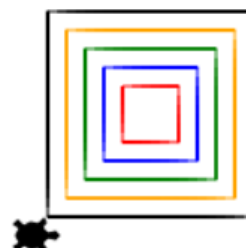
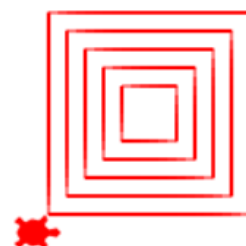
\*\*\*\*\*

## TEME:

- Crtanje vise istih figura
  - Naredbom for se poziva druga naredba for
- Tipovi informacija - liste i ntorke
  - Definicija
  - Pristup elementima
  - Slicnosti i razlike
- Objašnjavanje preko
  - Scratch (koordinatni sistem, biblioteke, olovka..)
  - Logo (koncept kretanja kornjace)

## PRIMER 3: Upisani kvadrati u razlicitoj boji

- V0
  - Jedan kvadrat koji se crta preko for naredbe
- V1
  - Dodamo spoljni for koji će da
    - Se vrti 5 puta
    - Se vrti po novoj promenljivoj
  - Pokrenemo to i vidimo da se iscrta pet kvadrata ali na istom mestu, a potrebno je da se iscrta stalno malo veci
- V2
  - Dodamo da se nakon untrasnjeg fora kornjaca prepozicionira niže i levo za 10 piksela kako bi se crtao novi kvadrat
  - Pokrenemo i vidimo da sada fali da se veličine kvadrata promene u svakoj iteraciji
- V3
  - Dodamo promenu dužine stranice kvadrata u unutrašnjem for-u
  - Svaki novi kvadrat je veći od prethodnog za po 10 piksela na obe strane
  - Potrebno je dodati onda uvećenje od 2\*10 u svakoj spoljnoj iteraciji
  - Izračunati par dužina stranica na tabli i kako se dobija kada se gleda vrednost promenljive j
- V4
  - Dodavanje razlicitih boja - lista imena boja
  - Pristup elementima nizu je od 0 do 4, kao sto su vrednosti za j



V0 - jedan kvadrat crvene boje

```
import turtle

turtle.shape("turtle")
turtle.color("red")
turtle.pendown()
for i in range(4):
    turtle.forward(30)
    turtle.left(90)
turtle.penup()
```

V1 - pet kvadrata crvene boje, na istom mestu, jedan preko drugog

```
import turtle

turtle.shape("turtle")
turtle.color("red")
turtle.pendown()
for j in range(5):
    for i in range(4):
        turtle.forward(30)
        turtle.left(90)
    turtle.penup()
```

V2 - pomeraj svakog novog kvadrata da se ne bi crtali jedan preko drugog, istih dimenzija

```
import turtle
turtle.shape("turtle")
turtle.color("red")
for j in range(5):
    turtle.pendown()
    for i in range(4):
        turtle.forward(30)
        turtle.left(90)
    turtle.penup()
    turtle.backward(10)
    turtle.right(90)
    turtle.forward(10)
    turtle.left(90)
```

V3 - racunanje svake nove stranice, ako je najmanja stranica 30, a svaka sledeca se uvecava za po 10 sa obe strane levo i desno, onda su velicine 30, 50, 70, ... sto moze da se napravi paralela da je taj od 30 piksela nulti kvadrat, ovaj od 50 prvi, od 70 drugi, itd

0 1 2 3 4 -> vrednosti za j

30 50 70 90 110 -> vrednosti stranica

-> izlazi da se moze napraviti formula,  $30 + 20 \cdot j$ , gde to 'nesto' predstavlja umnozак od j i broja 20. Zato imamo formulu  $30 + 20 \cdot j$  kao formulu za definisanje velicine stranice bilo kog kvadrata

```
import turtle
```

```
turtle.shape("turtle")
turtle.color("red")
for j in range(5):
    turtle.pendown()
    for i in range(4):
        turtle.forward(30 + 10*j*2)
        turtle.left(90)
    turtle.penup()
    turtle.backward(10)
    turtle.right(90)
    turtle.forward(10)
    turtle.left(90)
```

V4 - dodavanje liste boja, sa pet elemenata, jer imamo 5 kvadrata. Svaki element u toj listi ima svoju poziciju, svoj jedinstven broj, preko kog cemo svakoj liniji reci koju boju da koristi.

```
import turtle
turtle.shape("turtle")
boje = ["red", "blue", "green", "orange", "black"]
for j in range(5):
    turtle.color(boje[j])
    turtle.pendown()
    for i in range(4):
        turtle.forward(30 + 10*j*2)
        turtle.left(90)
    turtle.penup()
    turtle.backward(10)
    turtle.right(90)
    turtle.forward(10)
    turtle.left(90)
```

#### **NAPOMENA:**

- Objasniti listu u pythonu, sintaksu i pristup elementima
- Kod raditi inkrementalno
- Eventualno polazati i verziju sa randomom za boje