

Kornjačina grafika

Python

Modul turtle

- Naredba za pokretanje grafičkog modula turtle:

```
from turtle import *
```

Sadrži funkcije za crtanje

- Izvođenjem naredbi otvara se grafički prozor veličine 600x600 piksela
- Olovka (pokazivač) je postavljena u središtu prozora i usmjerena udesno
- Središte grafičkog prozora ima koordinatu(0,0)

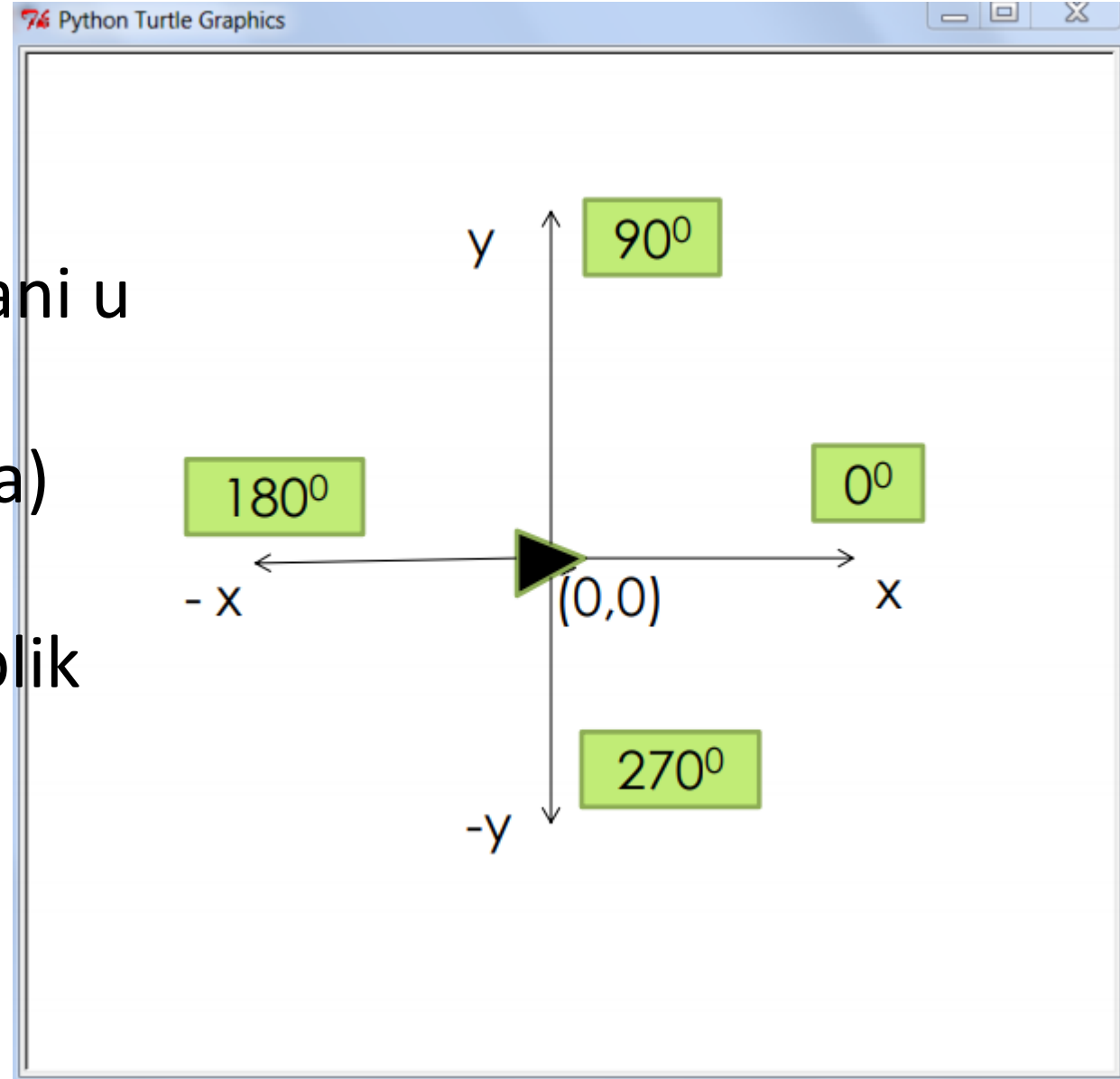
Svojstva olovke

Osnovna svojstva:

- Pozicija u prozoru (pomaci su zadani u pikselima)
- Orijentacija (zadana u stupnjevima)

Dodatna svojstva:

- Položaj (podignut ili spušten)
- Oblik
- Boja
- Širina



Pomicanje olovke - kornjače

Pomicanja se rade obzirom na trenutni položaj olovke

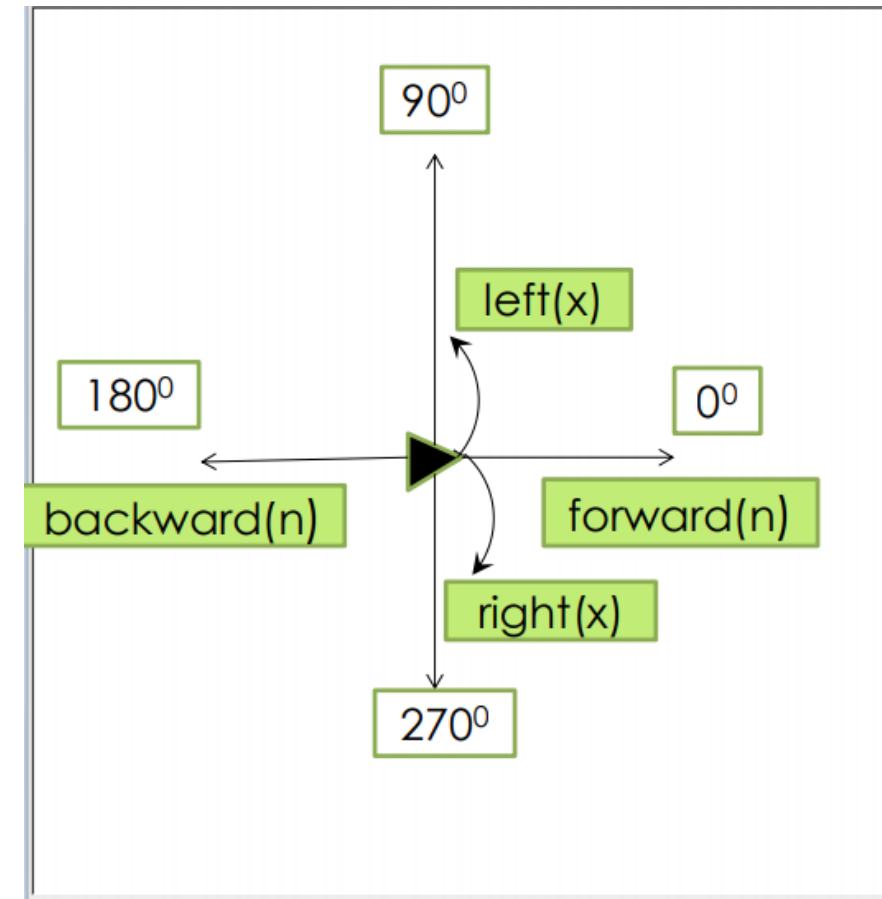
Naredba:

forward(n), **fd(n)** -Pomiče olovku ravno za n koraka u smjeru u kojem je trenutno okrenuta

backward(n), **bk(n)**, back(n) Vraća olovku natrag za n koraka

right(x), **rt(x)** „Zakreće“ olovku u smjeru smjera kazaljke na satu za x stupnjeva (x može biti realan broj)

left(x), **lt(x)** „Zakreće“ olovku u smjeru obrnutom od smjera kazaljke na satu za x stupnjeva



Ostale funkcije:

penup(), **pu()**, up() Podiže olovku iznad papira tako da ne ostavlja trag prilikom kretanja

pendown(), **pd()**, down() Vraća olovku na papir tako da sljedeće crtanje ostavlja trag

home() Vraća olovku na sredinu grafičkog ekrana (0, 0) te postavlja smjer olovke na 0 stupnjeva – ukoliko je olovka na papiru ova naredba ostavlja trag

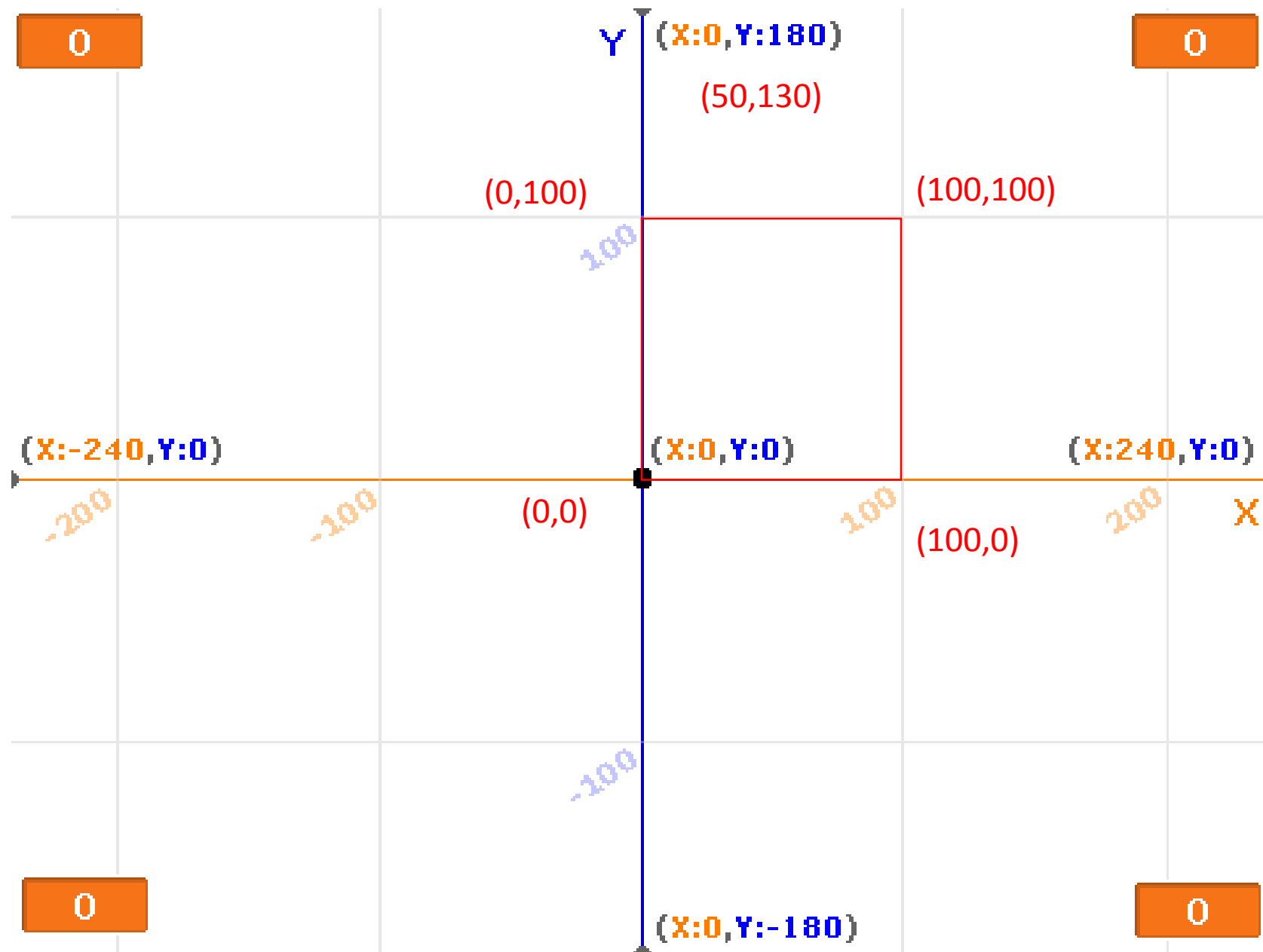
hideturtle(), **ht()** Skriva olovku za crtanje

showturtle(), **st()** Prikazuje olovku za crtanje

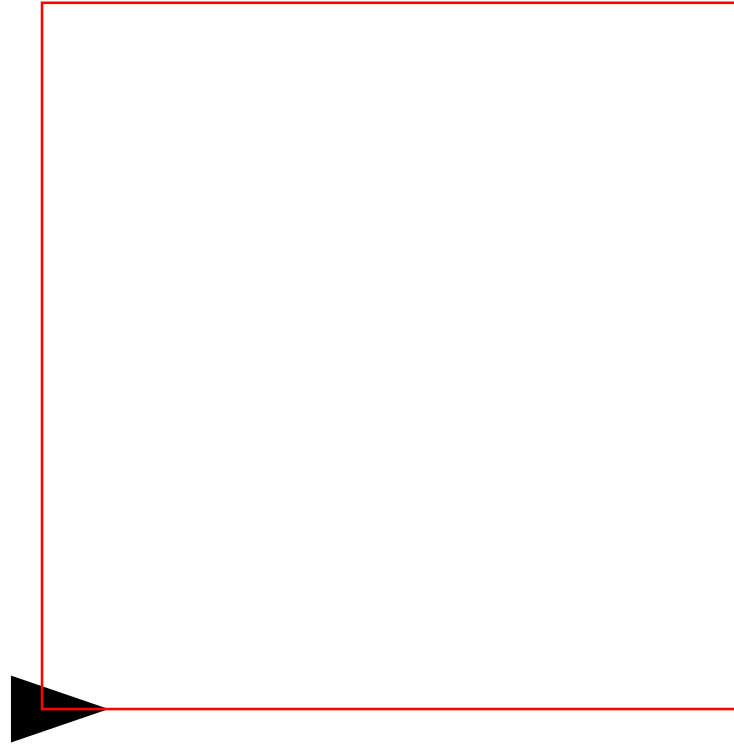
clear() Briše sadržaj grafičkog prozora

undo() Briše posljednji napravljeni korak

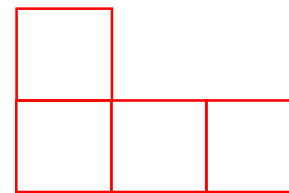
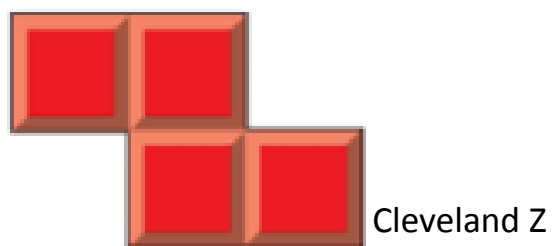
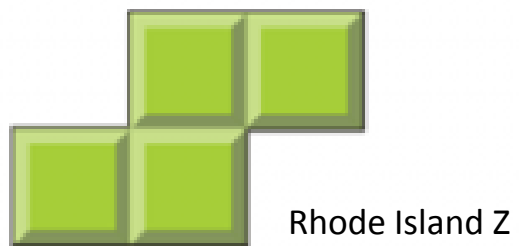
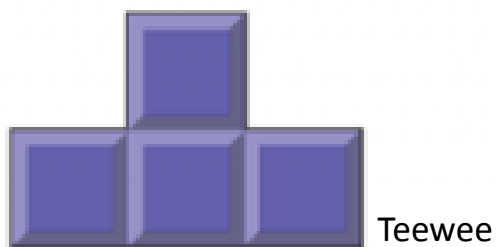
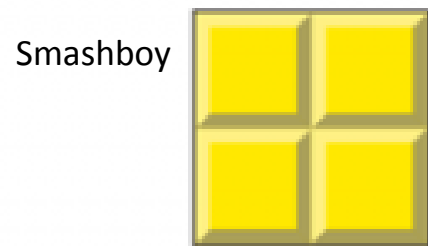
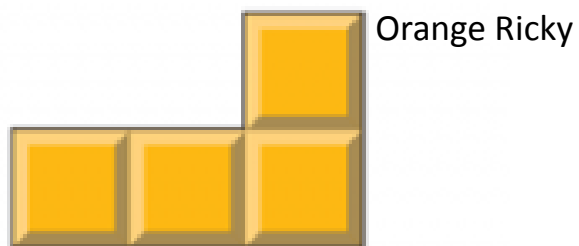
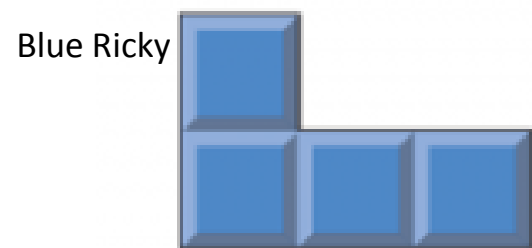
reset() Briše sve crteže i postavlja olovku u početni položaj



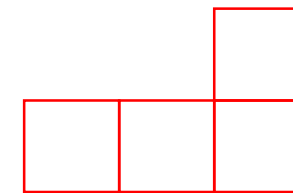
(100)



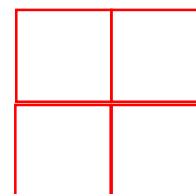
Tetris: - nacrtaj sve tetris blokove



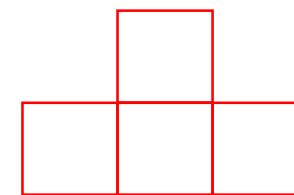
(50)



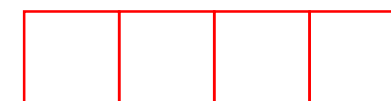
(50)



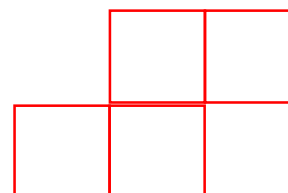
(50)



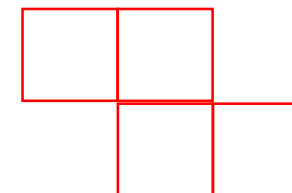
(50)



(50)



(50)



(50)

Algoritam ponavljanja – for petlja

Kornjačina grafika - Python

Algoritam slijeda - kvadrat

Pročitati !!!!

U dosadadšnjim primjerima kornjaču smo programirali algoritmom slijeda. Tako je kvadrat nacrtan algoritmom slijeda izgledao ovako:

```
from turtle import*
```

```
fd(100)
```

```
lt(90)
```

```
fd(100)
```

```
lt(90)
```

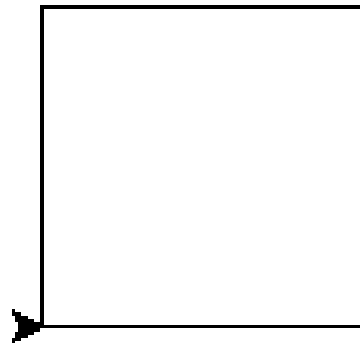
```
fd(100)
```

```
lt(90)
```

```
fd(100)
```

```
lt(90)
```

→ Ponavljanje naredbi



Uočavamo kako se 4 puta ponavljaju naredbe za kretanje kornjače fd (100) i za skretanje kornjače lt(90)

Kada imamo potrebu ponavljati radnje, tada je riječ o algoritmu ponavljanja koji se programira pomoću for petlje.

Algoritam ponavljanja – for petlja

```
for i in range(a):    a- broj ponavljanja
    naredba
```

For petlja omogućava ponavljanje naredbi određeni broj puta. Sve naredbe napisane unutar petlje (uvučene) ponavljaju se onoliko puta koliko zadamo parametrima petlje, u ovom slučaju to je a – broj ponavljanja

Prepisati u bilježnicu do tu !!!

Kvadrat iz prethodnog primjera nacrtan algoritmom slijeda izgledao bi ovako:

```
from turtle import*
for i in range(4):  → Parametar petlje – u slučaju kvadrata je 4, petlja se izvodi 4 puta
    fd(100)
    lt(90) } → naredbe koje se ponavljaju su uvučene unutar petlje
```

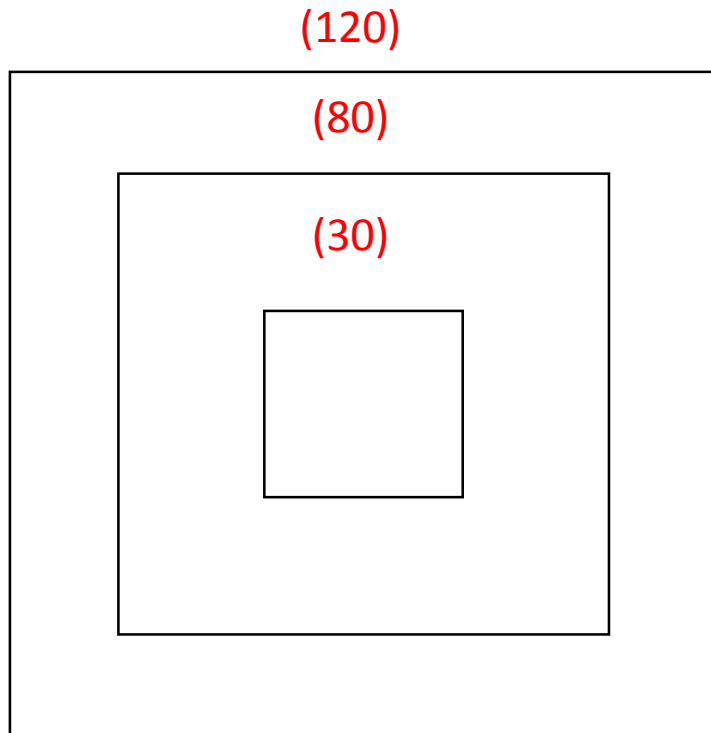
Odraditi u Pythonu!!!

Uvlaka nastaje nakon što napišemo naredbu `for i in range(4):` i pritisnemo enter.

Primjer korištenja for petlje: Kvadrati 1

Pročitati!!!

Ovaj smo primjer radili na satu algoritmom slijeda. Prvo smo crtali kvadrat duljine 120, zatim smo podigli kornjaču, pomaknuli je na točku crtanja drugog kvadrata, spustili kornjaču i zatim nacrtali kvadrat stranice duljine 80. Postupak smo ponovili i za kvadrat duljine 30.



Primjer korištenja for petlje: Kvadrati 1

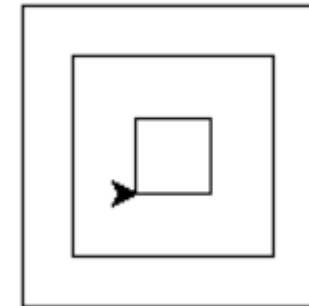
Algoritam slijeda:

```
from turtle import*
#kvadrat duljine stranice 120
fd(120)
lt(90)
fd(120)
lt(90)
fd(120)
lt(90)
fd(120)
lt(90)
#podizanje i pomak kornjače
pu()
fd(20)
lt(90)
fd(20)
rt(90)
pd()
#kvadrat duljine stranice 80
fd(80)
lt(90)
fd(80)
lt(90)
fd(80)
lt(90)
fd(80)
lt(90)
#podizanje i pomak kornjače
pu()
fd(25)
lt(90)
fd(25)
rt(90)
pd()
#kvadrat duljine stranice 30
fd(30)
lt(90)
fd(30)
lt(90)
fd(30)
lt(90)
fd(30)
lt(90)
```

Algoritam ponavljanja:

```
from turtle import*
#kvadrat duljine stranice 120
for i in range(4):
    fd(120)
    lt(90)
#podizanje i pomak kornjače
pu()
fd(20)
lt(90)
fd(20)
rt(90)
pd()
#kvadrat duljine stranice 80
for i in range(4):
    fd(80)
    lt(90)
#podizanje i pomak kornjače
pu()
fd(25)
lt(90)
fd(25)
rt(90)
pd()
#kvadrat duljine stranice 30
for i in range(4):
    fd(30)
    lt(90)
```

Odraditi u Pythonu!!!

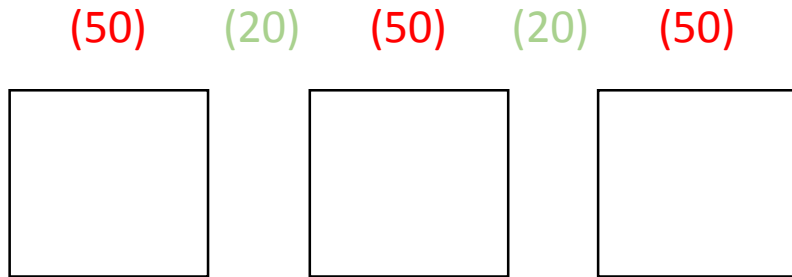


Koristeći se algoritmom ponavljanja, tj. for petljom smanjili smo broj naredbi u programu. Kod je pregledniji i zautima manje prostora u memoriji

Primjer korištenja for petlje: Kvadrati 2

Nacrtati kvadrate prema slici koristeći naredbu ponavljanja - for petlju. Zelena vrijednost označava udaljenost između kvadrata

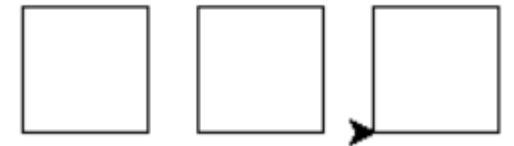
Odraditi u Pythonu!!!



Ovdje možemo pomoću tri for petlje nacrtati 3 kvadrata.

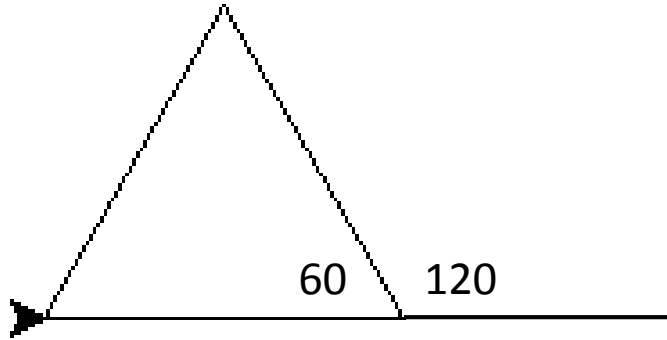
Naredbe za kretanje možemo pisati jednu ispod druge ili u jednom redu, ali ih onda odvajamo točka zarezom.

```
from turtle import*
#prvi kvadrat
for i in range(4):
    fd(50)
    lt(90)
#podizanje i pomak kornjače
pu()
fd(70)
pd()
#drugi kvadrat
for i in range(4):
    fd(50)
    lt(90)
#podizanje i pomak kornjače 2
pu();fd(70);pd()
#treći kvadrat
for i in range(4):
    fd(50)
    lt(90)
```



Primjer korištenja for petlje: Trokut

Ncrtati jednakostraničan trokut duljine stranice 100

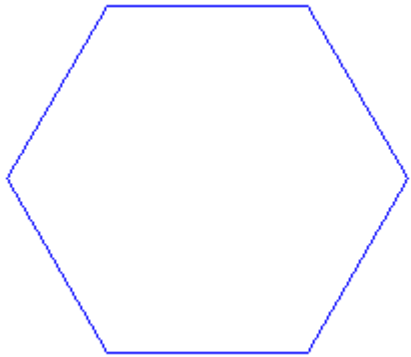


```
from turtle import*  
for i in range(3):  
    fd(100)  
    lt(120)
```

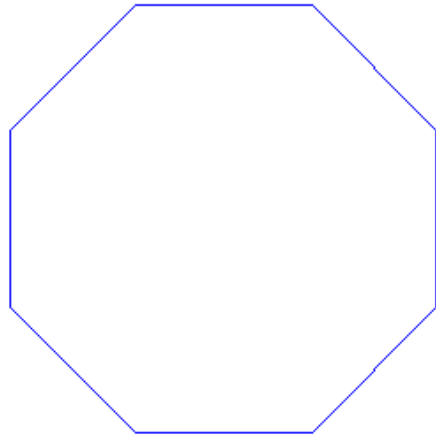
Vanjski kut iznosi 120 stupnjeva ($180 - 60$) i to je upravo ona vrijednost za koju Kornjača mora skrenuti

Primjer korištenja for petlje: Mnogokuti

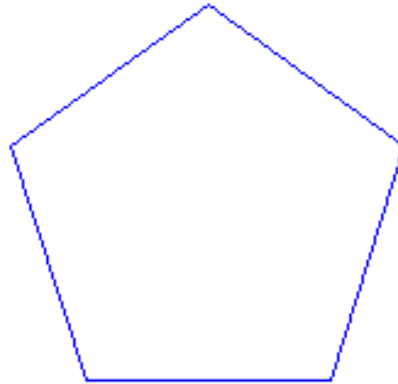
- Napravite programe koji crtaju 6-erokut, 8-erokut, 5-erokut. Duljina stranice svakog mnogokuta neka bude 50



6-erokut



8-erokut

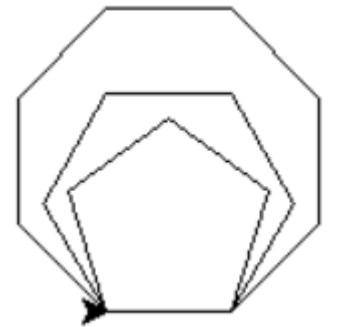


5-erokut

```
from turtle import*  
for i in range (6):  
    fd(50)  
    lt(60)
```

```
from turtle import*  
for i in range (8):  
    fd(50)  
    lt(45)
```

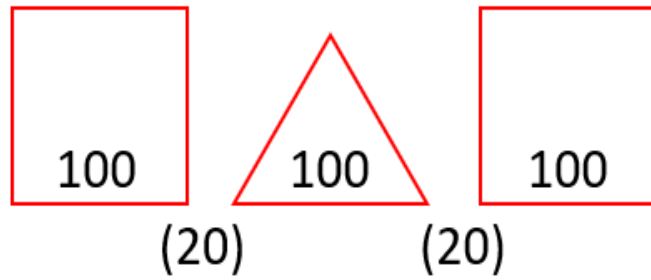
```
from turtle import*  
for i in range (5):  
    fd(50)  
    lt(72)
```



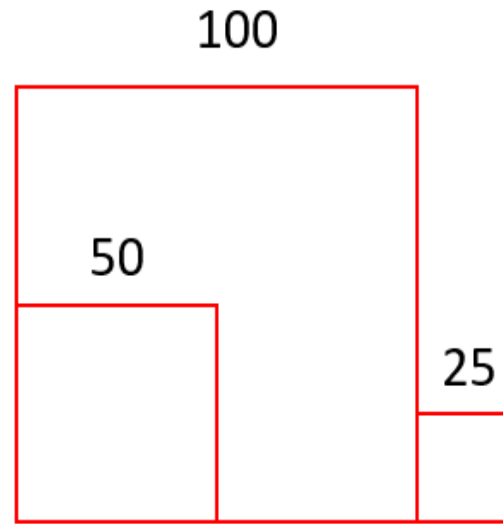
Kut skretanja = 360 / broj kuteva mnogokuta

Zadaci za vježbu: Svaki se crtež sastoji od nekoliko likova koje je potrebno nacrtati koristeći for petlju

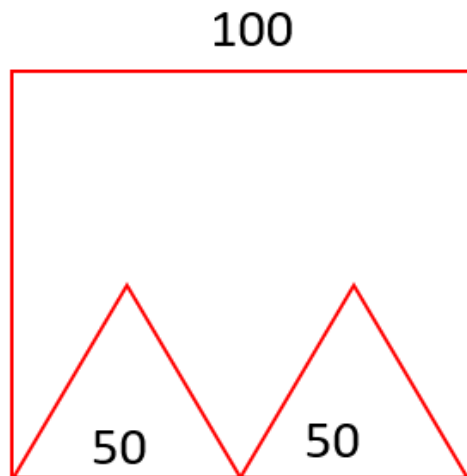
Crtež 1



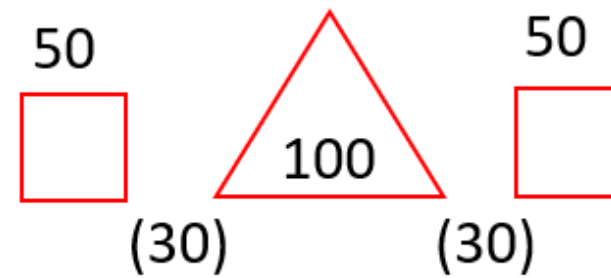
Crtež 2



Crtež 3

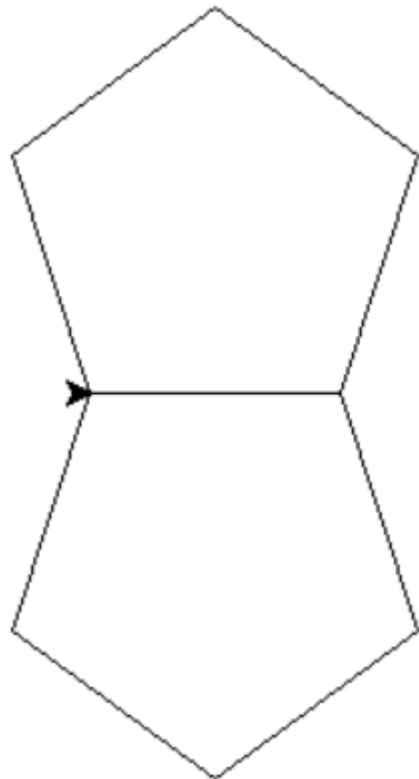


Crtež 3

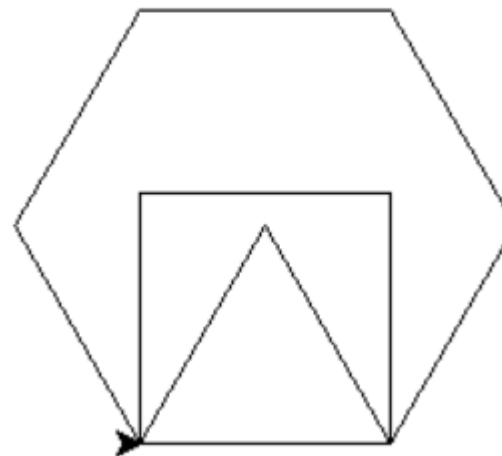


Zadaci za vježbu: Svaki se crtež sastoji od nekoliko likova koje je potrebno nacrtati koristeći for petlju (duljina stranice svih likova je 100)

Crtež 4



Crtež 5



Crtež 6

