



Extramaterial till Matematik Z

Algebra

ELEV

Det finns många olika programmeringsspråk. Ett av dem är Python, som du ska få bekanta dig med i de här uppgifterna när du ritar med Python och Turtle. Du kommer att få testa, tolka och skriva egen kod samt använda loopar för att rita stjärnor. Du kommer att få lära dig att använda slumpal för att slumpa fram koordinaterna för stjärnorna.

Uppgiften bygger vidare på Nivå ETT och TVÅ samt uppgifterna i ”Programmering och digital kompetens”, Algebra, Matematik Y. Se även ”Lathund – Python with turtle” för fler tips.

SYFTE

Syftet med övningen är att du ska

- få bekanta dig med programmeringsspråket Python 3.
- lära dig att använda en editor för att skriva program.
- få erfarenhet av att rita med Turtle.
- testa, tolka och skriva egen kod.
- förstå varför och hur man använder sig av loopar.
- förstå varför och hur man använder sig av funktioner.
- få erfarenhet av att använda villkor.
- få erfarenhet av hur man ger variabler värden och hur man kan förändra värdet.
- få förståelse för koordinatsystemets uppbyggnad och förstå hur koordinater för en viss punkt anges.
- få erfarenhet av kommandot goto(x,y)
- kunna slumpa fram värden på x och y och därmed slumpa fram stjärnornas placering.
- kunna planera och genomföra en egen programmeringsuppgift och skapa en färgglad stjärnhimmel.

REDOVISNING/BEDÖMNING

Du lämnar in kod, bilder och svar till din lärare.

DEL 1: Python with turtle - stjärnor och funktioner

RITA STJÄRNOR - UPPGIFT A

1. Börja med att gå in på repl.it och välj ”start coding” och sedan ”Python (with Turtle)” och ”Create Repl”.

För en djupare genomgång av editorn repl.it och dess funktioner, se ”Lathund – Python with Turtle”.

2. Skriv in koden:

```
from turtle import *
for i in range(5):
    forward(80)
    left(144)
```

Observera att den första raden (`from turtle import *`) alltid ska stå kvar, även i kommande övningar.

3. Kör koden genom att klicka på ”Run”. Vad händer?
4. Vad betyder din kod?
 - a) Vad står 100 respektive 144 för?
 - b) ”`for i in range(5):`” är en loop. Vad är det som ska upprepas och hur många gånger?

Koden vi angett är ett sätt att ange koden för att en stjärna med fem uddar ska ritas. Det finns även andra sätt, till exempel där man anger antalet uddar och räknar ut vinkeln turtle ska vrida sig genom att ta $180 - (180/\text{antalet uddar})$. Dock behöver man då korrigera för decimaltal mm så vi håller oss till just den här modellen av stjärnor.

SKAPA OCH ANROPA FUNKTIONER - UPPGIFT B

1. Skriv in koden (håll reda på indragen):

```
for i in range(5):
    forward(80)
    left(144)
goto(100,100)
for i in range(5):
    forward(100)
    left(144)
```

2. Kör koden och tolka den.

3. Istället för att skriva koden för hur en stjärna ritas varje gång man vill rita en stjärna, kan man **skapa en funktion** som heter "rita_stjarna" (eller till exempel "draw_star")

Då måste man börja med att definiera vad funktionen innebär. Det gör man så här:

```
def rita_stjarna():  
    for i in range(5):  
        forward(100)  
        left(144)
```

När man sedan vill rita en stjärna, behöver man bara **anropa funktionen** genom att skriva `rita_stjarna()`

Skriv in och kör koden:

```
def rita_stjarna():  
    for i in range(5):  
        forward(100)  
        left(144)  
    rita_stjarna()
```

4. Fungerade funktionen?
5. Vad tror du kommer att hända om du lägger till en rad med funktionen `rita_stjarna()`? Testa!

```
rita_stjarna()  
rita_stjarna()
```

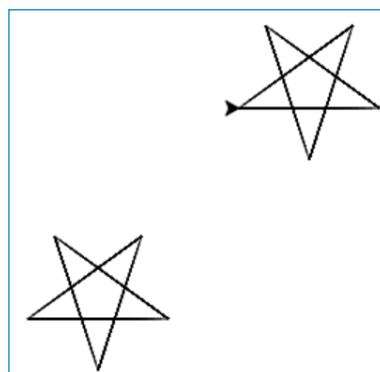
6. Lägg till en förflyttning, `goto(100,100)`, mellan de två funktionerna och kör programmet.

```
rita_stjarna()  
goto(100,100)  
rita_stjarna()
```

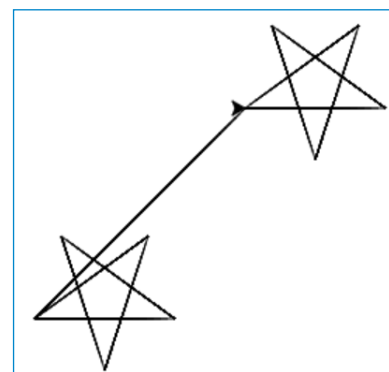
7. För att "snygga till" bilden kan man använda sig av kommandot `penup()` respektive `pendown()`. Fundera själv ut var dessa två kommandon ska placeras för att det inte ska ritas ett streck mellan stjärnorna.

TIPS: Hur skulle du gjort om du använde papper och penna?

så här:



istället för:



8. Testa programmet på egen hand genom att göra tillägg och/eller förändringar.

Exempel på användbara kommandon:

<code>speed(5)</code>	Snabbar upp hastigheten på Turtle
<code>color("pink")</code>	Ändrar färg på "pennan"
<code>screen().bgcolor("blue")</code>	Ändrar bakgrundsfärg
<code>begin_fill()</code> och <code>end_fill()</code>	Fyller figuren med färg:

```
begin_fill()
rita_stjarna()
end_fill()
```

DEL 2: Python with turtle - stjärnor och slump

Hur gör man om man vill rita en stjärnhimmel med 10, 50 eller kanske 100 stjärnor?

Som du kanske märkte i uppgiften innan, är det ganska tidsödande att skriva in alla positioner (koordinater) som man vill placera stjärnorna på.

Därför ska du nu låta Python hjälpa dig att slumpa fram värden på x och y till kommandot `goto(x,y)` och därmed alltså slumpa fram placeringar för stjärnorna.

1. För att kunna hantera slumpantal måste du börja med att importera `randint` (som slumpar fram heltal) från biblioteket `random`:

```
from turtle import *
from random import randint
```

2. Skriv/kopiera och klistra därefter in följande kod. Håll reda på stavning, skiljetecken, parenteser och indrag!

```
speed(5)
def draw_star():
    for i in range(5):
        forward(50)
        left(144)
num_stars=0
while num_stars < 5:
    x = randint(-200, 200)
    y = randint(-200, 200)
    draw_star()
    penup()
    goto(x,y)
    pendown()
    num_stars=num_stars+1
```

Nu ser din kod,
förhoppningsvis, ut så här:

```
1 from turtle import *
2 from random import randint
3
4 speed(5)
5
6 def draw_star():
7     for i in range(5):
8         forward(50)
9         left(144)
10
11 num_stars=0
12
13 while num_stars < 5:
14     x = randint(-200, 200)
15     y = randint(-200, 200)
16     draw_star()
17     penup()
18     goto(x,y)
19     pendown()
20     num_stars=num_stars+1
```

3. Kör programmet och notera vad som händer.
4. Nu är det dags att försöka tolka koden, bit för bit.

Ett tips är att skriva kommentarer i koden. Det gör man genom att använda tecknet #. Se exempel:

```
1 from turtle import *
2 from random import randint #gör så att vi kan slumpa fram heltal
3
4 speed(5) #anger hastigheten på "pennan"
5
```

Försök själv. Skriv en eller flera kommentarer till följande rader med kod:

```
6 def draw_star():
7     for i in range(5):
8         forward(50)
9         left(144)
```

5. Vilket värde får variabeln num_stars innan loopen startar? Varför?

```
11 num_stars=0
```

6. Den sista delen innehåller en loop med ett villkor (while). Hur länge/många gånger ska loopen upprepas?

```
12 while num_stars < 5:
13     x = randint(-200, 200)
14     y = randint(-200, 200)
15     draw_star()
16     penup()
17     goto(x,y)
18     pendown()
19     num_stars=num_stars+1
```

7. I nästa del används **randint**. Vad är det som slumpas fram?

```
12 while num_stars < 5:
13     x = randint(-200, 200)
14     y = randint(-200, 200)
15     draw_star()
16     penup()
17     goto(x,y)
18     pendown()
19     num_stars=num_stars+1
```

8. Vad händer om man ändrar talen i parenteserna efter "randint", exempelvis till (0, 200)?
9. Tolka följande delar av looperna, steg för steg. Vad är det som ska hända? Vad är x och y?

```
12 while num_stars < 5:
13     x = randint(-200, 200)
14     y = randint(-200, 200)
15     draw_star()
16     penup()
17     goto(x,y)
18     pendown()
19     num_stars=num_stars+1
```

10. Looperna avslutas med att värdet på variabeln **num_stars** ändras. Varför? Vad är värdet av variabeln efter att man kört looperna första gången?

DEL 3: Python with turtle – Eget arbete

1. Använd de kunskaper du har från tidigare uppgifter för att skapa ett eget konstverk. Det kan till exempel vara en mörkblå stjärnhimmel med 100 stjärnor i olika storlekar och färger.
Använd dig av loopar, slump och villkor.
2. Presentera ditt konstverk och din kod med tillhörande kommentarer som förklarar vad som händer i de olika stegen.