

# Lab02

CSGE601020

25 September 2020

Dasar – Dasar Pemrograman 1

## Menggambar dengan Modul Turtle - 2

### Gambaran Umum Tugas

*Baca dokumen tugas ini dengan seksama sebelum membuat program.*

Lab ini akan berfokus dalam menghasilkan berbagai macam *warna* dan beberapa bentuk bangun sederhana menggunakan *library* dari Python. Kalian akan menggunakan konsep pengulangan (*for loop*) dan juga *simultaneous assignment statements* dalam lab kali ini.

### Tugas

#### Bagian A

Kalian akan tetap menggunakan Turtle Graphics seperti pada Lab01.

1. Gunakan statement *for* untuk menggambar **poligon dengan 5 sisi** (pentagon).
2. Tambahkan sebuah fitur pada program untuk mengisi warna pada poligon. Warna ini didapatkan dari kombinasi warna **merah**, **hijau**, dan **biru** yang diinput oleh user.
3. Gunakan statement *for* untuk menggambar **spiral pentagon**. Kalian dapat melihat contohnya di bawah.

Perlu diingat bahwa *Turtle graphics* adalah **graphics package** 2 dimensi yang menggunakan sistem koordinat Kartesius. Kalian bisa bayangkan ada sebuah kura-kura (*turtle*) yang memiliki pena di badannya. Kura-kura ini dapat bergerak dan menggambar ke segala arah yang kamu instruksikan. Titik **(0,0)** koordinat kartesius ada tepat di tengah *turtle window*. Modul turtle dalam Python ini sebenarnya adalah hasil pengimplementasian “perilaku” dari program turtle graphics yang orisinil (untuk info mengenai program turtle yang orisinil, dapat kalian cek di [https://en.wikipedia.org/wiki/Turtle\\_graphics](https://en.wikipedia.org/wiki/Turtle_graphics)). Di dalam IDLE (shell), setelah kalian meng-*import* turtle (**import turtle**), kalian dapat menggunakan perintah **help(turtle)** untuk mendapatkan informasi lebih lanjut mengenai modul *turtle*.

Alur program yang akan kalian buat adalah sebagai berikut:

1. Meminta *user* untuk memasukkan komponen warna. Terdiri dari **R (red/merah)**, **G (green/hijau)**, **B (blue/biru)**. Buatlah warna dari ketiga komponen ini.
2. Gambar dan beri warna pentagon menggunakan statement **for**. Posisi dari pentagonnya adalah di sebelah kiri terhadap *turtle window*.
3. Setelah menggambar no.2, pindah ke posisi selanjutnya, yaitu posisi kanan dari *turtle window*, dan gambarlah sebuah **spiral pentagon** dengan menggunakan statement **for** dengan warna yang sudah diberikan oleh *user*.

### Informasi Tambahan

Method	Deskripsi
<code>turtle.penup()</code>	Mengangkat pena (menonaktifkan mode menggambar).
<code>turtle.pendown()</code>	Menurunkan pena (mengaktifkan mode menggambar).
<code>turtle.forward( jarak )</code>	Memajukan <i>turtle</i> sejauh ( <i>jarak</i> ) dan mengarah sesuai dengan arah gerak <i>turtle</i> .
<code>turtle.backward( jarak )</code>	Memundurkan <i>turtle</i> sejauh ( <i>jarak</i> ) dan mengarah sesuai dengan arah gerak <i>turtle</i> .
<code>turtle.right( derajat )</code> , <code>turtle.left( derajat )</code>	Mengganti/membelokkan arah gerak turtle ke <b>kanan/kiri</b> sebesar ( <b>derajat</b> ).
<code>turtle.goto(x, y)</code>	Memindahkan <i>turtle</i> ke titik yang spesifik ( <b>x,y</b> ). Jika pena masih dalam keadaan <i>down</i> maka dalam perpindahannya turtle akan tetap menggambar, jika pena dalam keadaan <i>up</i> , maka <i>turtle</i> tidak akan menggambar. Catatan: <i>Turtle</i> akan selalu berada pada posisi (0,0) ketika <i>window turtle</i> terbuka.
<code>turtle.color(r,g,b)</code>	Mengatur warna dari pena yang akan digunakan. Warna dari pena tidak akan berubah selama kamu tidak mengubahnya. Method ini membutuhkan 3 argumen, yaitu <b>r (red/merah)</b> , <b>g (green/hijau)</b> , dan <b>b (blue/biru)</b> . Ketiga argument tersebut menerima bilangan desimal antara 0.0 sampai 1.0.
<code>turtle.begin_fill()</code> , <code>turtle.end_fill()</code> :	Gunakan perintah <b>“turtle.begin_fill()”</b> sebelum kamu memulai menggambar sesuatu, kemudian gambar, setelah itu akhiri dengan perintah <b>“turtle.end_fill()”</b> . Figur yang kamu gambar di antara kedua perintah tersebut akan diisi dengan warna yang kamu atur.

<code>turtle.textinput(title, prompt)</code>	Menampilkan sebuah jendela <i>pop up</i> dengan sebuah judul ( <i>title</i> ) yang bisa kamu atur dan sebuah <i>prompt</i> dengan format string. Fungsi dari method ini adalah meminta input dari <i>user</i> .
<code>turtle.bye()</code>	Menutup <i>window</i> /jendela turtle.
<code>turtle.hideturtle()</code>	Membuat <i>turtle</i> tidak terlihat.
<code>turtle.shape(bentuk)</code>	Mengatur bentuk dari <i>turtle</i> . Opsi yang tersedia adalah 'arrow', 'turtle', 'circle', 'square', 'triangle', 'classic'.
<code>turtle.setheading(angle)</code>	Mengatur orientasi sesuai ( <i>angle</i> ) yang diberikan.
<code>turtle.exitonclick()</code>	Membuat program otomatis tertutup ketika <i>user</i> memencet click pada <i>turtle window</i> .

## Poligon Beraturan

Rumus derajat total semua sudut di dalam poligon beraturan dengan  $n$  sisi adalah  $180 * (n - 2)$ , jadi setiap sudut interiornya adalah  $180 * (n - 2) / n$ .

Dari situ, kamu dapat menghitung seberapa jauh harus berbelok untuk setiap **sudut luar**:  
 $180 - 180 * (n - 2) / n = 360 / n$ .

Gunakan statement **for** untuk menggambar poligon beraturan dengan  $n$  sisi.

## Pengulangan

Metode pengulangan **for** akan sangat berguna ketika kalian tahu seberapa banyak kalian akan melakukan repetisi. Contoh: kalian tahu berapa banyak sisi poligon yang akan kalian gambar, maka kalian dapat menerapkan **for**. Lihat [Slide 01](#) untuk melihat contoh dari penggunaan **for**.

## Mencampur warna

Program kalian akan membaca **nilai warna** yang telah *user* masukkan dan kalian dapat menggunakan *method* yang tersedia pada *turtle* untuk menggambar poligon dengan nilai warna yang sudah diberikan oleh *user*. Sebenarnya, ada banyak cara untuk membuat sebuah warna, tetapi cara yang paling sering digunakan di dalam pemrosesan grafis komputer adalah dengan memproses *additive color* (referensi: [http://en.wikipedia.org/wiki/Additive\\_color](http://en.wikipedia.org/wiki/Additive_color)), yaitu dengan mengkombinasi berbagai macam warna **merah**, **hijau**, dan **biru** dapat menciptakan sebagian besar warna yang ada.

Contohnya, berikut adalah cara menggambar sebuah lingkaran berwarna kuning: 1 red + 1 green + 0 blue. Ketikkan perintah berikut satu per-satu di dalam IDLE dan lihat apa yang terjadi. Kemudian coba modifikasi, jika nilainya diubah menjadi 0.3 red + 0.7 green + 0.6 blue, warna apa yang kalian dapat?

```
>>> import turtle
>>> turtle.color(1,1,0)
>>> turtle.begin_fill()
>>> turtle.circle(80) # draw a circle with radius = 80 units
>>> turtle.end_fill()
>>> turtle.hideturtle()
>>> turtle.bye()
```

### Selesaikan program berikut!

Ikuti alur program di bawah dengan melengkapi pada bagian yang rumpang dengan mengikuti petunjuk yang tersedia pada komentar.

```
-----
# Author: .....
# File name: lab02a.py
# menggunakan turtle untuk menggambar pentagon dan spiral pentagon
# meminta input komponen warna (r,g,b) dari user

import turtle
turtle.shape('turtle')
turtle.title('Lab 02')

# mengambil nilai dari komponen warna merah yang diinput user
r = float(turtle.textinput(
    "Lab 02", "The red color component [between 0 and 1]: "))

# mengambil nilai dari komponen warna hijau yang diinput user
g = .....
```

```

# mengambil nilai dari komponen warna biru yang diinput user
b = .....

# membuat warna dari kombinasi nilai rgb yang diberi user
turtle.color(.....)

# memindahkan turtle ke sebelah kiri turtle window
# dan mengatur headingnya menjadi 0
.....
.....
.....
.....

# gambar dan beri warna pentagon dengan warna
# yang sudah diatur di atas
# gunakan for loop
.....
.....
.....
.....

# geser turtle ke sebelah kanan di turtle window
# dan atur headingnya menjadi 0
.....
.....
.....
.....

# gambar sebuah spiral pentagon
side = 150
for i in range(.....):
    turtle.forward(side)
    turtle.right(72) # sudut luar dari pentagon adalah 72 derajat

```

```

        side = ..... # kurangi variabel side dengan 2
# membuat turtle tidak terlihat
turtle.hideturtle()

# pesan untuk user
turtle.up()
turtle.goto(0, 200)
turtle.down()
turtle.color('blue')
turtle.write("Please click on the graphics window to exit ...",
            False, align='center', font=('Arial', 20, 'normal'))

# tunggu user mengclick screen untuk
# mengexit window turtle

turtle.exitonclick()
turtle.mainloop()

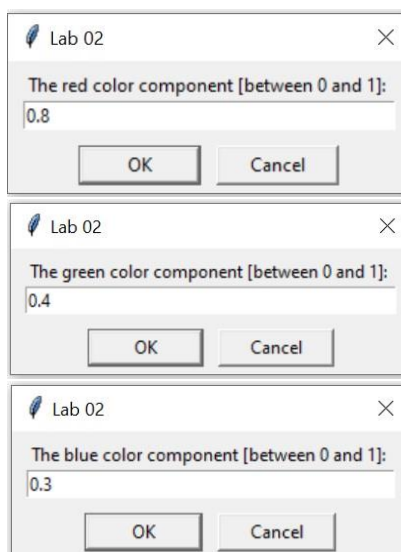
# the end

```

---

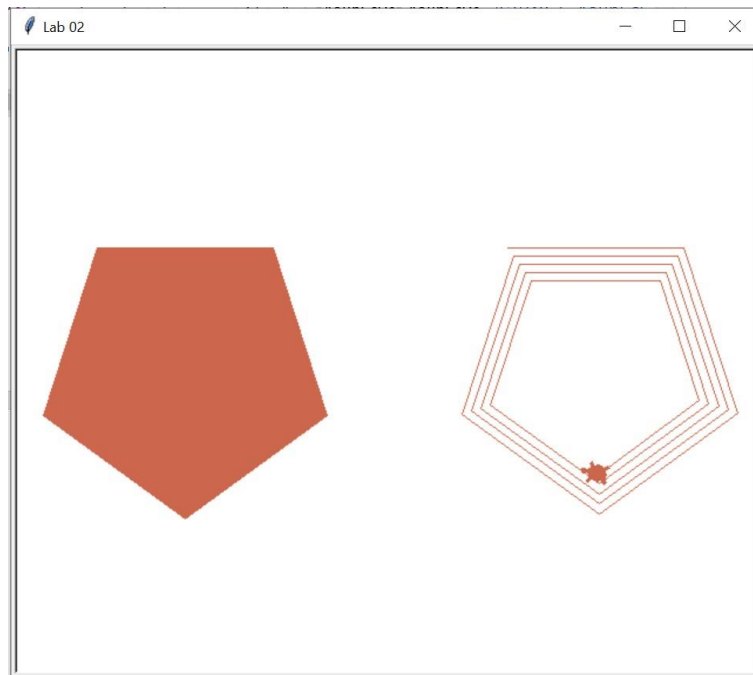
## Contoh

Input dari user

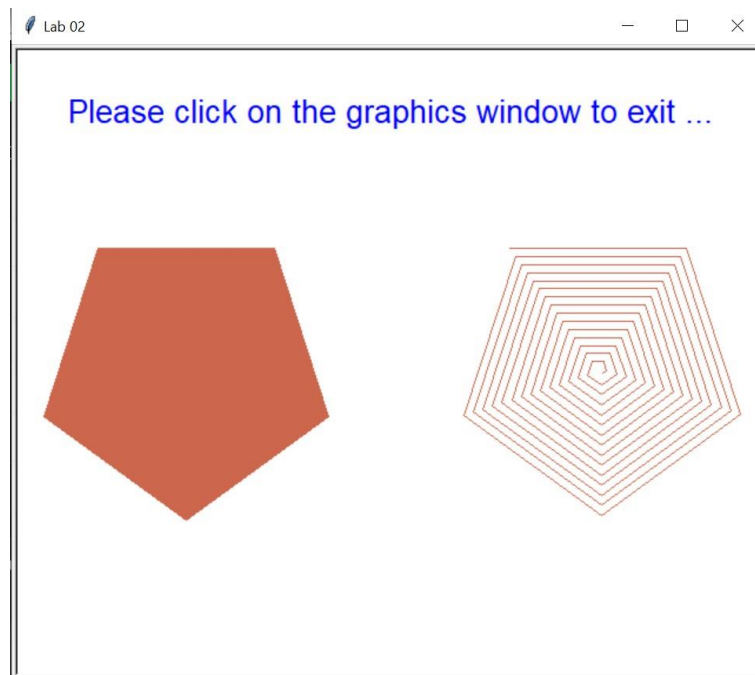


The image shows three stacked dialog boxes, each titled 'Lab 02'. The first dialog box asks for 'The red color component [between 0 and 1]:' and has the input field containing '0.8'. The second dialog box asks for 'The green color component [between 0 and 1]:' and has the input field containing '0.4'. The third dialog box asks for 'The blue color component [between 0 and 1]:' and has the input field containing '0.3'. Each dialog box has 'OK' and 'Cancel' buttons at the bottom.

Proses menggambar



Gambar selesai



## Bagian B

Untuk mengisi program yang rumpang berikut, kalian harus menulis beberapa baris ***simultaneous assignment statement*** dalam bentuk ***x,y = y,x*** untuk mengurutkan nilai yang ditunjuk oleh variabel **number1, number2, number3, number4, number5**, sehingga ketika di-*print*, nilai mereka sudah terurut dari yang paling kecil hingga paling besar. **3 perintah *simultaneous assignment*** cukup untuk mengurutkan variabel-variabel tersebut.

```
# Author: .....
# File name: lab02b.py
# menggunakan simultaneous assignment untuk;
# mengurutkan data

# 5 variabel dengan nilai sbb:
number1 = 25
number2 = 7
number3 = 3
number4 = -8
number5 = 19

# output yang diminta: 25 7 3 -8 19
print("Initial data: ")
print(number1, number2, number3, number4, number5)

# 3 simultaneous assignment cukup untuk
# menukar nilai dari variabel yang ada;
# secara bersamaan, dengan bentuk: x,y = y,x
.....
.....
.....

# Menampilkan data yang telah terurut
# # output yang diminta: -8 3 7 19 25
print("Sorted data, from smallest to largest: ")
print(number1, number2, number3, number4, number5)
```



## File yang Disubmit

Kumpulkan file python kalian (lab02a.py dan lab02b.py) dalam bentuk **zip** dengan penamaan sebagai berikut:

**"<kode\_asdos>\_<nama\_kamu>\_<NPM>\_lab02.zip"**

Contoh: "FA\_TonyStark\_2006285610\_lab02.zip" dan kumpulkan di SCeLE.

Pastikan program kalian **terdokumentasi dengan rapi, dan dapat dijalankan.**

**That's all for today. Happy programming! 'Met ngoding!**

**L. Y. Stefanus & The Asdos Team**