

Lab02

CSGE601020

10 September 2021

Dasar – Dasar Pemrograman 1

Menggambar dengan Modul Turtle

Gambaran Umum Tugas

Baca dokumen tugas ini seluruhnya dengan seksama sebelum membuat program.

Lab ini akan berfokus dalam menghasilkan berbagai macam *warna* dan beberapa bentuk geometri sederhana menggunakan *library* turtle graphics dari Python. Kalian akan menggunakan konsep pengulangan (***for loop***) dalam lab kali ini.

Turtle graphics adalah ***graphics package*** 2 dimensi yang menggunakan sistem koordinat cartesian. 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 cartesian ada tepat di tengah *turtle window*. Modul **turtle** dalam Python adalah hasil pengimplementasian dari turtle graphics (untuk info mengenai turtle graphics, dapat kalian cek di https://en.wikipedia.org/wiki/Turtle_graphics). Di dalam IDE IDLE atau Spyder, setelah kalian meng-*import* turtle (**`import turtle`**), kalian dapat menggunakan perintah **`help(turtle)`** untuk mendapatkan informasi lebih lanjut mengenai modul *turtle*.

Tugas

Alur program yang akan kalian buat adalah sebagai berikut:

1. Mintalah pengguna untuk memasukkan komponen warna: R (komponen merah), G (komponen hijau), dan B (komponen biru). Buatlah warna dari ketiga komponen ini menggunakan perintah **color(R, G, B)**. Nilai yang digunakan untuk tiap komponen warna adalah nilai float di dalam interval [0.0, 1.0]
2. Mintalah kepada pengguna banyaknya sisi poligon yang akan digambar. Simpanlah input user ini dalam sebuah variabel, misalnya **n**. Asumsikan bahwa berlaku **$3 \leq n \leq 10$**
3. Gambarlah dan warnailah bagian dalam poligon tersebut menggunakan warna yang telah diberikan oleh pengguna. Gunakanlah perintah **for** ketika menggambar poligon tersebut. Posisi poligon tersebut berada ditengah-tengah window turtle. Panjangnya poligon berbanding terbalik terhadap n, yakni $500 // n$.

Turtle commands (methods)

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>) dimana arah pergerakannya sesuai dengan arah gerak <i>turtle</i> .
<code>turtle.backward(jarak)</code>	Memundurkan <i>turtle</i> sejauh (<i>jarak</i>) dimana arah pergerakannya sesuai dengan arah gerak <i>turtle</i> .
<code>turtle.right(derajat)</code> , <code>turtle.left(derajat)</code>	Mengganti/membelokkan arah gerak turtle ke kanan/kiri sebesar (derajat) relatif terhadap arah sebelum dibelokkan.
<code>turtle.goto(x, y)</code>	Memindahkan <i>turtle</i> ke titik yang spesifik (x,y). Jika pena masih dalam keadaan <i>down</i> maka selama berpindah, turtle akan tetap menggambar. Jika pena dalam keadaan <i>up</i> , maka <i>turtle</i> tidak akan menggambar. Catatan: <i>Turtle</i> berada pada posisi (0,0) ketika <i>window turtle</i> terbuka pertama kali.
<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 r (red/merah), g (green/hijau), dan b (blue/biru). Ketiga argument tersebut menerima <i>floating number</i> antara 0.0 sampai 1.0.
<code>turtle.begin_fill()</code> , <code>turtle.end_fill()</code>	Gunakan perintah <i>turtle.begin_fill()</i> sebelum kamu memulai menggambar sesuatu, kemudian menggambarlah. Setelah selesai menggambar, akhiri dengan perintah <i>turtle.end_fill()</i> . 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 (title) yang bisa kamu atur dan sebuah prompt dengan format string. Fungsi dari method ini adalah meminta input dari <i>user</i> dalam tipe str.
<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, relatif terhadap sumbu x positif. Arah yang dihasilkan bersifat absolut, yakni tidak terpengaruh oleh arah sebelum perintah
<code>turtle.exitonclick()</code>	Membuat program menunggu sampai <i>user</i> click pada <i>turtle window</i> dan menutup program secara otomatis.

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**, yaitu $180 - 180 * (n - 2) / n = 360 / n$.

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

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), yaitu dengan mengkombinasi tiga warna dasar: **merah**, **hijau**, dan **biru**, yang ternyata dapat menciptakan sebagian besar warna yang ada.

Contoh: cara untuk membuat warna **kuning** adalah `turtle.color(1,1,0)`, yaitu menggunakan **1 red** + **1 green** + **0 blue**.

Ketikkan perintah-perintah berikut satu per-satu di dalam shell python dan lihat apa yang terjadi. Bagaimana jika perintah `turtle.begin_fill()` dan `turtle.end_fill()` dihapus?

```
import turtle

# mari memulai Turtle Graphics window dengan kura-kura merah
# Perintah alternatif: turtle.color('red')
turtle.color(1,0,0)

# mengatur bentuk dari turtle
turtle.shape('turtle')

# menurunkan pen sehingga semua pergerakan akan ditunjukkan
# sebagai garis merah
turtle.pendown()

# Semua gambar yang digambar antara perintah
# turtle.begin_fill() dan turtle.end_fill()
# akan diisi dengan warna saat ini
turtle.begin_fill()

# bergerak ke depan sejauh 100 unit sesuai arah saat ini
turtle.forward(100)

# memutar arah turtle sebesar 120 derajat
turtle.right(120)

# bergerak kedepan sebesar 100 unit sesuai arah saat ini
turtle.forward(100)

# memutar arah turtle sebesar 120 derajat
turtle.right(120)

# bergerak kedepan sebesar 100 unit sesuai arah saat ini
turtle.forward(100)

# Semua gambar yang digambar antara perintah
# turtle.begin_fill() dan turtle.end_fill()
# akan diisi dengan warna saat ini
turtle.end_fill()

# menunggu pengguna menekan layar untuk keluar
turtle.exitonclick()
```

Contoh interaksi dengan user pada saat input

Lab 02

Komponen warna merah [antara 0 and 1]:

0.8

OK Cancel

Lab 02

Komponen warna hijau [antara 0 and 1]:

0.4

OK Cancel

Lab 02

Komponen warna biru [antara 0 and 1]:

0.3

OK Cancel

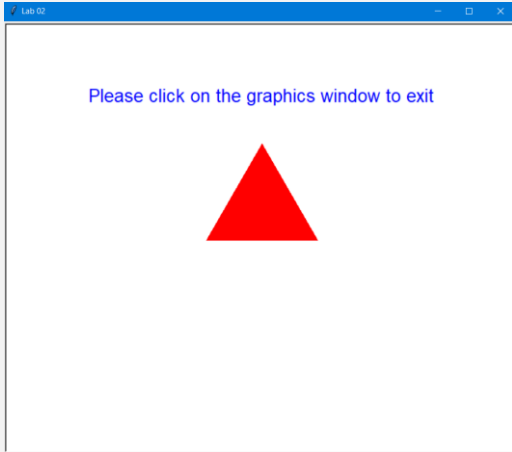
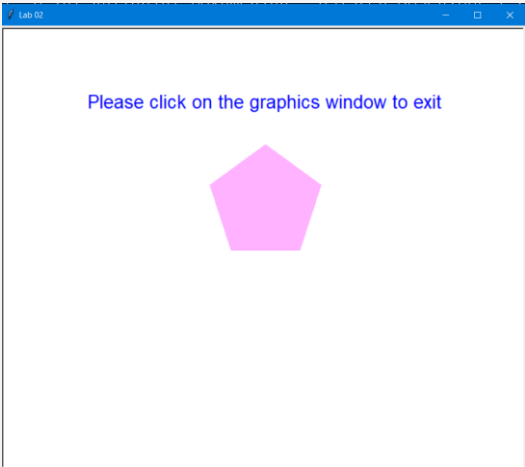
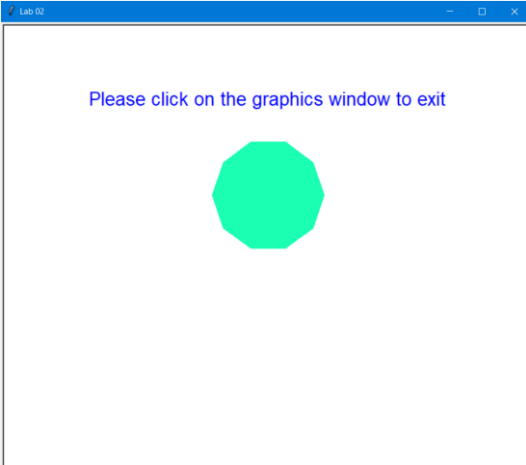
Lab 02

Masukkan jumlah sisi pada poligon [antara 3 dan 10]:

3

OK Cancel

Contoh-contoh output dari program

Red	Green	Blue	n	Final display of <i>turtle</i> window
1.0	0	0	3	
1.0	0.7	1.0	5	
0.1	1.0	0.7	10	

Selesaikan program berikut!

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

```
-----
# Nama: .....
# lab02.py
# menggunakan turtle untuk menggambar poligon
# meminta input komponen warna (r,g,b) dari user, serta jumlah sisi pada poligon

import turtle

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

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

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

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

# meminta input n, jumlah sisi poligon yang ingin kita gambar
n = ...

# membuat warna dari kombinasi nilai rgb yang diberi user
...

# default panjang sisi luar pentagon
DEFAULT_SIDE_LEN = 500

# sudut dalam dari polygon
internal_angle = 360 / n

# Menyesuaikan panjang dari sisi poligon relatif terhadap n
side_len = DEFAULT_SIDE_LEN // n

# memindahkan turtle ke posisi baru pada bagian tengah
# window turtle. Setelah itu, aturlah arahnya dengan tepat
turtle.penup()
turtle.goto(... , 0)
...

# gambarlah sebuah poligon dengan warna yang telah diatur diatas,
# kemudian isilah warna didalamnya dengan warna yang sama.
# gunakanlah for loop
...
...

...
...
...

turtle.end_fill()
turtle.penup()

# membuat turtle menjadi tak terlihat
turtle.hideturtle()

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

# menunggu pengguna untuk meng-click pada layar untuk keluar
turtle.exitonclick()
# selesai
-----
```

File yang Disubmit

Kumpulkan program kalian dalam sebuah file python dengan penamaan sebagai berikut:

“<kode_asdos>_<nama_kamu>_<NPM>_lab02.py”

Gunakanlah Snake_Case tanpa menyertakan spasi pada bagian nama.

Contoh: “FA_TonyStark_2006285610_lab02.py” dan kumpulkan di SCellE.

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

Rubrik untuk lab ini adalah sebagai berikut:

Total marks : 100

Kebenaran program (Test cases)	90
Komentar yang jelas	10

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

L. Y. Stefanus & The Asdos Team