

LAPORAN PRAKTIKUM 3

Diajukan untuk memenuhi salah satu tugas praktikum Mata kuliah Komputer Grafik



Disusun Oleh:

Benny Yoga Suhardi (211511035)

**JURUSAN TEKNIK KOMPUTER DAN INFORMATIKA
PROGRAM STUDI D-3 TEKNIK INFORMATIKA
POLITEKNIK NEGERI BANDUNG
2022**

DAFTAR ISI

DAFTAR ISI.....	2
Pertanyaan 6 Atribute Bentuk Dasar	3
Pertanyaan 7 Attribute garis dengan Custom Line	6
Lesson Learn.....	7
CURHAT BEBAS.....	8
REFERENSI	8

Pertanyaan 6 Atribut Bentuk Dasar

Buatlah bentuk Dasar sbb:

- Persegi
- Persegi Panjang
- Segitiga Siku-Siku
- Trapesium Siku-Siku dan Trapesium Sama Kaki
- Jajar Genjang

Jawaban

- Untuk yang persegi saya menggunakan line dda pada keempat sisinya dengan dimana parameter dari function nya dari square adalah square(vector, Panjang , warna) disini saya menggunakan warna karena untuk di kombinasikan dengan bangun datar lainnya seperti trapezium dan lainnya dan dinisi untuk setiap line dimasukkan kedalam array dan kemudian di append untuk setiap perubahan dan diganti sesuai dengan titik kordinatnya untuk lebih jelas dalam Source Code Berikut

```
10 func square(titik_awal : Vector2, panjang_sisi, warna_kanan, warna_kiri):
11   var res = PoolVector2Array()
12   res.append_array([lineDDA(titik_awal.x, titik_awal.y, titik_awal.x + panjang_sisi, titik_awal.y, Color.aqua)])
13   res.append_array([lineDDA(titik_awal.x + panjang_sisi, titik_awal.y, titik_awal.x + panjang_sisi, titik_awal.y + panjang_sisi, warna_kanan)])
14   res.append_array([lineDDA(titik_awal.x + panjang_sisi, titik_awal.y + panjang_sisi, titik_awal.x, titik_awal.y + panjang_sisi, Color.aqua)])
15   res.append_array([lineDDA(titik_awal.x, titik_awal.y + panjang_sisi, titik_awal.x, titik_awal.y, warna_kiri)])
16   return res
17
```

- Untuk persegi Panjang kurang lebih sama seperti persegi Panjang menggunakan Line DDA pada keempat sisinya tetapi perbedaannya disini menggunakan 4 parameter pada functionnya dimana parameternya yaitu Rectangle(Vector, Panjang, Lebar, Warna) untuk lebih jelas dari penggunaannya pada Source code berikut

```
17
18 func rectangle(titik_awal : Vector2, panjang_sisi, lebar_sisi):
19   var res = PoolVector2Array()
20   res.append_array([lineDDA(titik_awal.x, titik_awal.y, titik_awal.x + panjang_sisi, titik_awal.y, Color.aqua)])
21   res.append_array([lineDDA(titik_awal.x + panjang_sisi, titik_awal.y, titik_awal.x + panjang_sisi, titik_awal.y +
22     + lebar_sisi, Color.aqua)])
23   res.append_array([lineDDA(titik_awal.x + panjang_sisi, titik_awal.y + lebar_sisi, titik_awal.x, titik_awal.y +
24     + lebar_sisi, Color.aqua)])
25   res.append_array([lineDDA(titik_awal.x, titik_awal.y + lebar_sisi, titik_awal.x, titik_awal.y, Color.aqua)])
26   return res
```

- Pada segitiga siku siku dibutanya sama dengan persegi menggunakan Line DDA dimana di dalam segitiga hanya menghubungkan 3 vertex menggunakan Line DDA tersebut, disini

menggunakan 4 parameter dari penggunaan functionnya yaitu

Tringale(Vector,Tinggi,Alas,Color,) color disini untuk pada bagian tinggi dimana pada bagian tinggi bis akita kombinasikan untuk bangun datar lainnya

```
27
28 v func tringale(titik_awal:Vector2, alas, tinggi, warna_tinggi):
29 >| var res = PoolVector2Array()
30 >| res.append_array([lineDDA(titik_awal.x,titik_awal.y,titik_awal.x,titik_awal.y + tinggi,warna_tinggi)])
31 >| res.append_array([lineDDA(titik_awal.x,titik_awal.y + tinggi ,titik_awal.x + alas ,titik_awal.y
32 >| + tinggi,Color.aqua)])
33 >| res.append_array([lineDDA(titik_awal.x + alas,titik_awal.y + tinggi,titik_awal.x,titik_awal.y,Color.aqua)])
34
```

- Trapesium siku siku dan trapesium sama kaki disini menggunakan gabungan antar persegi dan juga segitiga dimana untuk trapesium siku-siku pada bagian kanan dari persegi diberi warna transparan begitupun pada bagian tinggi dari segitiga diberi warna transparan dan untuk yang trapesium sama kaki kurang lebihnya sama penggabungan dari 1 persegi Panjang dan 2 segitiga siku siku yang dibalik untuk segitiga pada bagian kanan

```
34
35 v func trapesium(titik_awal:Vector2,panjang, alas):
36 #| square(titik_awal,panjang)
37 >| var res = PoolVector2Array()
38 >| square(titik_awal,panjang,Color.transparent,Color.aqua)
39 >| titik_awal.x = titik_awal.x+panjang
40 >| tringale(titik_awal,alas,panjang,Color.transparent)
41
42 v func trapesium_samakaki(titik_awal : Vector2,panjang,alas):
43 >| var res = PoolVector2Array()
44 >| square(titik_awal,panjang,Color.transparent,Color.transparent)
45 >| res.append_array([lineDDA(titik_awal.x,titik_awal.y + panjang,titik_awal.x - alas,titik_awal.y + panjang,Color.aqua)
46 >| res.append_array([lineDDA(titik_awal.x - alas,titik_awal.y + panjang,titik_awal.x,titik_awal.y,Color.aqua)])
47 >| titik_awal.x = titik_awal.x+panjang
48 >| tringale(titik_awal,alas,panjang,Color.transparent)
49
```

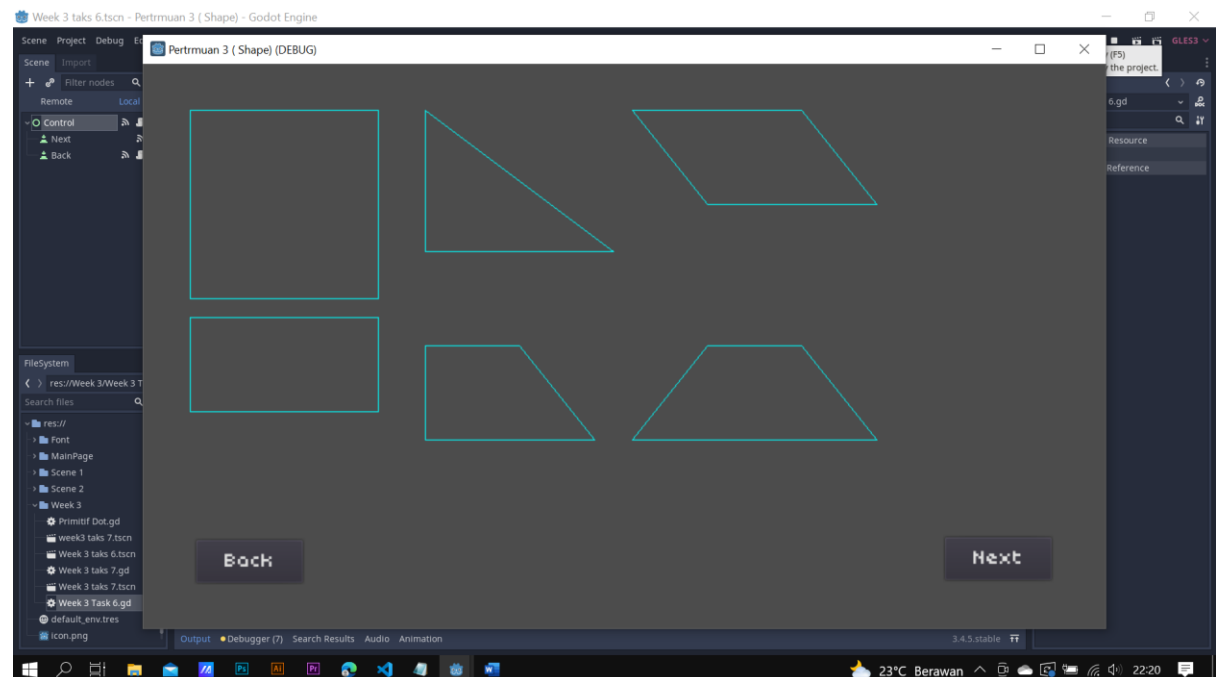
- Pada jajargenjang disini sama seperti trapesium sama kaki perbedaannya disini pada bagian segitiga pada bagian kanan dimana untuk yang trapesium di lakukan flip terhadap horizontal sedangkan pada jajargenjang dilakukan rotasi tapi rotasi 90 derajat ini dilakukan secara manual sehingga tanpa menggunakan method rotasi yang telah disediakan oleh godot

```
50 v func jajargenjang(titik_awal: Vector2, panjang,alas):
51 >| var res = PoolVector2Array()
52 >| square(titik_awal,panjang,Color.transparent,Color.transparent)
53 >| res.append_array([lineDDA(titik_awal.x,titik_awal.y,titik_awal.x - alas,titik_awal.y,Color.aqua)])
54 >| res.append_array([lineDDA(titik_awal.x - alas,titik_awal.y,titik_awal.x,titik_awal.y + panjang,Color.aqua)])
55 >| titik_awal.x = titik_awal.x+panjang
56 >| tringale(titik_awal,alas,panjang,Color.transparent)
57
```

Dan pada function node Drawnya dipanggil fungsi fungsi tersebut dengan mengatur ukuran ukurannya dan juga titik point koordinatnya dengan membuat variable barunya dari setiap koordinat untuk lebih jelas pada bagian berikut

```
0 func _on_Control_draw():
1   >| var square_point = Vector2(50,50);
2   >| var rectangle_point = Vector2(50,270)
3   >| var tringale_point = Vector2(300,50)
4   >| var trapesium_point = Vector2(300,300)
5   >| var trapesium_samakaki_point = Vector2(600,300)
6   >| var jajargenjang_point = Vector2(600,50)
7   >|
8   >| square(square_point,200,Color.aqua,Color.aqua)
9   >| rectangle(rectangle_point, 200, 100)
10  >| tringale(tringale_point,200,150,Color.aqua)
11  >| trapesium(trapesium_point,100,80)
12  >| trapesium_samakaki(trapesium_samakaki_point,100,80)
13  >| jajargenjang(jajargenjang_point,100,80)
14  >|
15  >| pass # Replace with function body.
```

Dan hasil jadi dari tak 6 tersebut akan menjadi seperti ini



Pertanyaan 7 Attribute garis dengan Custom Line

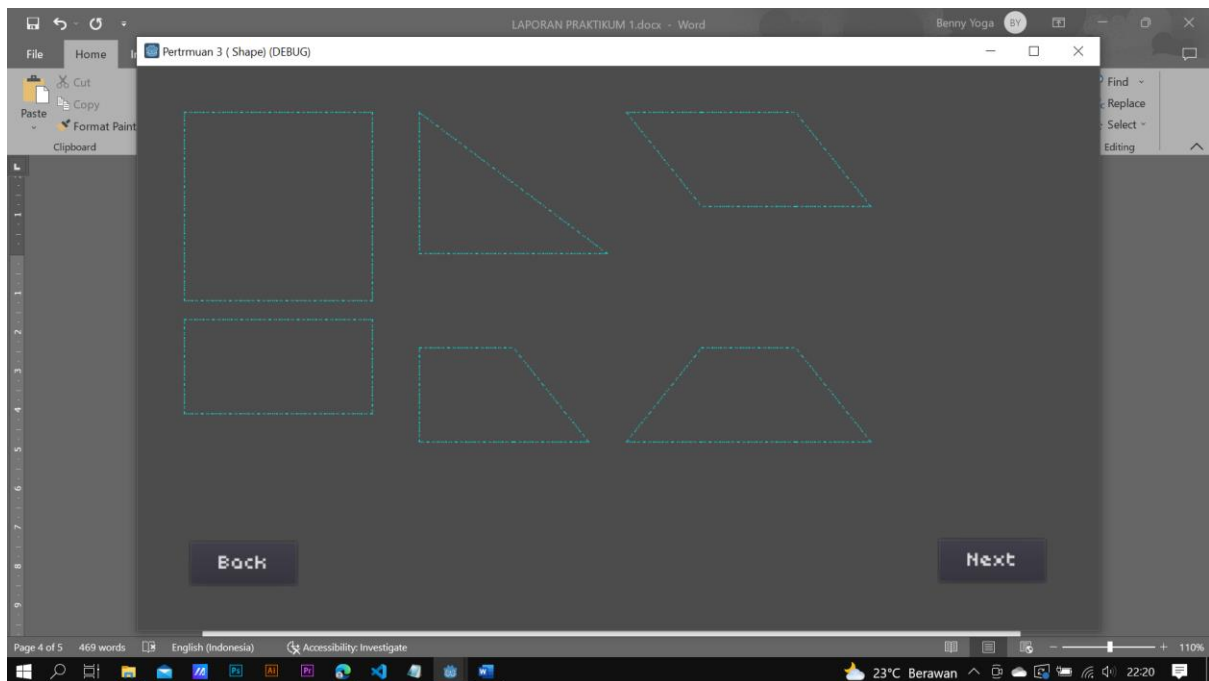
Buatlah bentuk Dasar sbb:

- Persegi
- Persegi Panjang
- Segitiga Siku-Siku
- Trapesium Siku-Siku dan Trapesium Sama Kaki
- Jajar Genjang
- Modifikasi hasil pertanyaan no 6 dengan fungsi garis yang telah dimodifikasi.

Untuk pertanyaan 7 ini saya menggunakan function yang sama dengan yang taks 6 tersebut perbedaan nya dari pembentuk garisnya sama memodifikasi garisnya langsung pada file primitive.gd yaitu pada bagian function LineDDA dimana saya memeberikan warna tranparan Ketika sebuah kondisi tertentu seolah olah akan menghasilkan garis yang putus putus pada line dda tersebut untuk lebih jelas dari modifikasinya pada source Code Berikut

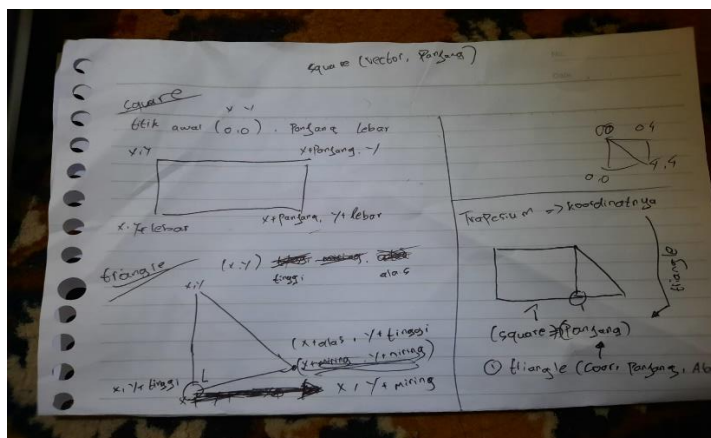
```
0 >|
1 ✓ func lineDDA(xa, ya, xb, yb, color):
2 >|     var dx = xb - xa
3 >|     var dy = yb - ya
4 >|     var steps
5 >|     var xIncrement
6 >|     var yIncrement
7 >|     var x = xa
8 >|     var y = ya
9 ✓ >|     if (abs(dx) > abs(dy)) :
10 >|         steps = abs(dx)
11 ✓ >|     else :
12 >|         steps = abs(dy)
13 >|
14 >|     xIncrement = dx/ float(steps)
15 >|     yIncrement = dy/ float(steps)
16 >|     put_pixel(round(x), round(y), color)
17 >|
18 ✓ >|     for k in steps:
19 >|         x += xIncrement
20 >|         y += yIncrement
21 ✓ >|         if(k%5 == 0):
22 >|             put_pixel(round(x), round(y), color)
23 ✓ >|         elif(k%2 == 0 || k%3==0):
24 >|             put_pixel(round(x), round(y), Color.transparent)
25 ✓ >|         else :
26 >|             put_pixel(round(x), round(y), color)
27
```

Untuk hasil dari modifikasinya maka akan menjadi seperti ini



Lesson Learn

Mungkin yang akan saya highlight pada lesson learn kali adalah dengan merencanakan atau mengkonsep terlebih dahulu apa yang ingin dikerjakan karena sebelumnya saya langsung saya memulai pekerjaan tanpa mengkonsep seperti apa dan pada kali ini saya mengkonsep terlebih dahulu untuk pembuatan dari setiap bangun datar pada kerta dan ternyata pada saat eksekusi tidak begitu bingung untuk apa yang dibuat



Dan lesson learn pada program yang saya dapatkan adalah dengan penggunaan array apalagi pada append dari setiap membuat line per line , vector 2d untuk menentukan dari titik per titik dalam sebuah variable dan juga penggunaan function di dalam function

CURHAT BEBAS

Assalamualaikum pak godot saya kemasukan roh goib sepertinya Ketika saya memulai untuk week 3 malah yang mulai itu di week 2 tetapi setelah keluar dari week 3 yang muncul week 2 dan masuk lagi akan masuk ke week 3 saya sudah menguliknya tetapi tetap seperti itu 😊

REFERENSI

Linietsky, J., 2014-2022. *GD script Engine*. [Online]

Available at:

https://docs.godotengine.org/en/stable/tutorials/scripting/gdscript/gdscript_basics.html

[Accessed 14 9 2022].

Linietsky, J., 2014-2022. *Godot Engine*. [Online]

Available at: https://docs.godotengine.org/en/3.0/classes/class_poolvector2array.html

[Accessed 2022 09 14].

Linietsky, J., 2020. *Godot Engine Q&E*. [Online]

Available at: <https://godotengine.org/qa/60939/transform-rotated-how-do-i-get-it-to-work>

[Accessed 2022 09 14].