

LAPORAN PRAKTIKUM 1

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

Task 1,2,3

Dalam kasus ini ada beberapa hal diantaranya untuk kasus no 1 kemudian 2 dan 3

Kasus 1

- Buatlah Bingkai / Margin secara static (masukan variable di Godot) atau Dinamis (gunakan export variable pada godot)
- Contoh Bidang canvas memiliki lebar(width) 840px dan tinggi (height) 480px
- Buatlah bingkai 10px
- Dapatkan koordinat pembangun bingkai (titik A, B, C, D) A ~ kiri atas, B kanan atas, C kiri bawah, dan D kanan bawah.
- Lalu disambungkan menggunakan algoritma pengembangan garis seperti lineDDA atau Bersenham.

Pembahasan

Untuk kasus no 1 disini saya menggunakan kedua algoritma yaitu bersenham dan juga Line DA untuk membentuk suatu border dimana awal mula dari membuat border tersebut adalah dengan mencari width dan juga height dari windowsnya kemudian dimasukkan kedalam variable setelah itu dipanggil sebuah prosedur dengan 4 parameter yaitu (x,y) (x',y') dan dari variable tadi yang telah dicari akan dijadikan sebagai pembatas dari border tersebut untuk lebih jelasnya dalam algoritma berikut

```
1 extends "res://Scene 1/Primitif.gd";
2
3 var width = ProjectSettings.get_setting("display/window/size/width");
4 var height = ProjectSettings.get_setting("display/window/size/height");
5 var center_width = (width)/2;
6 var center_height = (height)/2
7
8 func put_pixel(x, y, color):
9     draw_primitive(PoolVector2Array([Vector2(x, y)]), PoolColorArray([color]),PoolVector2Array(
10
11 func border():
12     lineBres(10,10,width-10,10,Color.aqua);
13     lineDDA(width-10,10,width-10,height-10,Color.aqua);
14     lineBres(width-10,height-10,10,height-10,Color.aqua);
15     lineDDA(10,height-10,10,10,Color.aqua);
16
17 func kartesian():
18
19     lineBres(10,center_height,width-10,center_height,Color.white);
20     lineDDA(center_width,height-10,center_width,10,Color.white);
21
```

Kasus 2

Buatlah Garis kartesian

- Contoh Bidang canvas memiliki lebar(width) 840px dan tinggi (height) 480px
- Buatlah cartesian
- Dapatkan koordinat pembangun kartesian dengan mendapatkan titik tengah dari width dan height.
- Lalu disambungkan menggunakan algoritma pengembangan garis seperti lineDDA atau Bresenham untuk membangun kartesian.

Dalam algoritma diatas untuk width dan juga height nya dikurangi dengan 10 px sebagai pembatas dan juga membentuk sebuah bordernya dan yang untuk kartesian digunakan variable yang berbeda yaitu center_width center_height sebagai penentu titik tengah dari garis horizontal dan vertical

```
1  extends "res://Scene 1/Primitif.gd";
2
3  var width = ProjectSettings.get_setting("display/window/size/width");
4  var height = ProjectSettings.get_setting("display/window/size/height");
5  var center_width = (width)/2;
6  var center_height = (height)/2
7
8  func put_pixel(x, y, color):
9      draw_primitive(PoolVector2Array([Vector2(x, y)]), PoolColorArray([color]), PoolVector2Array(
10     >| >|
11  func border():
12      >| lineBres(10,10,width-10,10,Color.aqua);
13      >| lineDDA(width-10,10,width-10,height-10,Color.aqua);
14      >| lineBres(width-10,height-10,10,height-10,Color.aqua);
15      >| lineDDA(10,height-10,10,10,Color.aqua);
16
17  func kartesian():
18      >|
19      >| lineBres(10,center_height,width-10,center_height,Color.white);
20      >| lineDDA(center_width,height-10,center_width,10,Color.white);
21
```

Kasus 3

Buatlah tiga fungsi geometric sembarang, dengan ketentuan sbb:

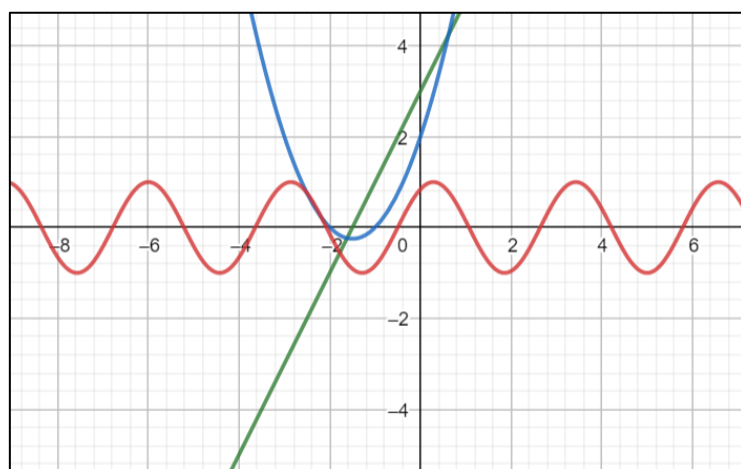
- Fungsi linear
- Fungsi kuadrat
- Fungsi trigonometri
- Letakan fungsi tersebut pada garis kartesian yang telah dibuat.

Dalam Task 3 ini diperlukan 3 fungsi yang berbeda dan saya membuat 3 buah fungsi tersebut untuk Fungsi linear menggunakan persamaan $2X + 3$ sedangkan untuk persamaan kuadratnya saya

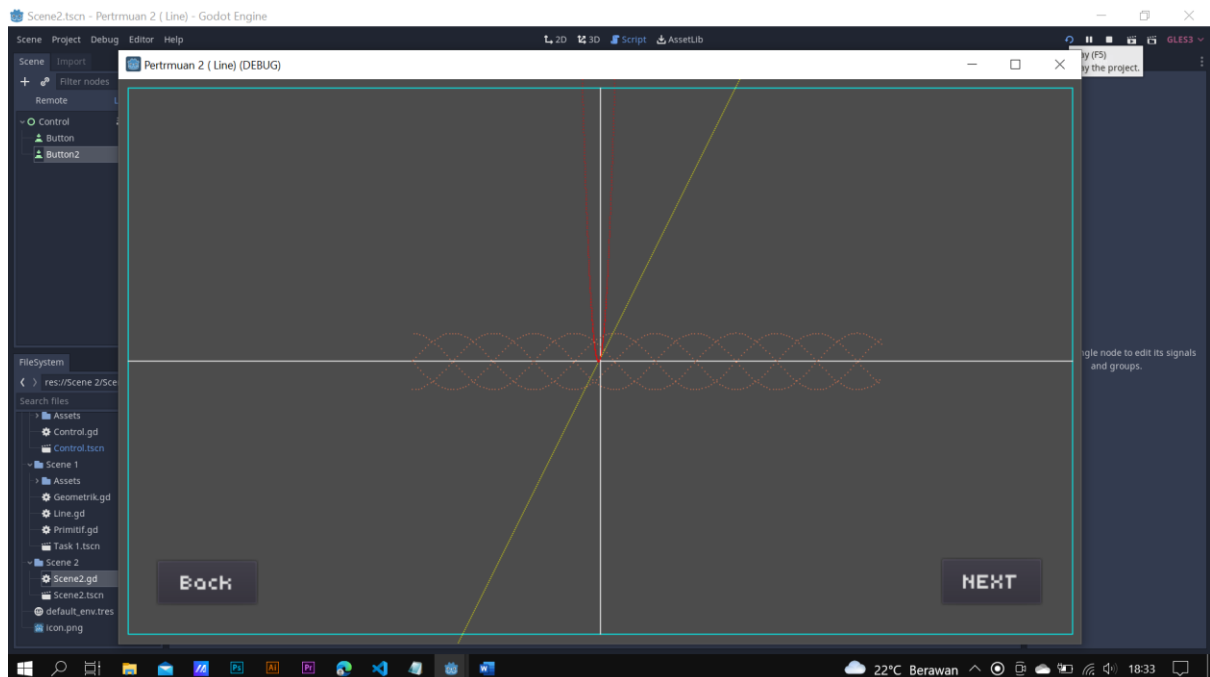
menggunakan persamaan $x^2 + 3x + 2$ dan yang terakhir untuk persamaan fungsi trigonometrinya saya menggunakan persamaan $\sin(2x+1)$ untuk persamaan dalam godot engine nya menjadi seperti ini dan juga menggunakan algoritma untuk mengeprint perpixel untuk membuat garis sesuai dengan persamaan tersebut

```
22 ▾ func linear():
23   ▸ var x = -200
24   ▸ var warna = Color.yellow;
25   ▸
26 ▾ ▸ while(x <center_height):
27   ▸   ▸ var y = 2*x + 3
28   ▸   ▸ put_pixel(x+center_width, center_height-y, warna)
29   ▸   ▸ x += 1;
30
31 ▾ func kuadrat():
32   ▸ var x = -200
33   ▸ var warna = Color.red;
34   ▸
35 ▾ ▸ while(x <center_height):
36   ▸   ▸ var y = x*x + 3*x + 2
37   ▸   ▸ put_pixel(x+center_width, center_height-y, warna)
38   ▸   ▸ x += 0.1;
39
40 ▾ func trigonometri():
41   ▸ var x = -200
42   ▸ var warna = Color.coral;
43   ▸
44 ▾ ▸ while(x <center_height):
45   ▸   ▸ var y = sin(2*x+1)*30
46   ▸   ▸ put_pixel(x+center_width, center_height-y, warna)
47   ▸   ▸ x += 0.8;
```

Untuk menggambarkan apa yang akan terjadi maka dapat dicoba menggunakan geogebra bagaimana persamaan tersebut akan tergambar maka akan menjad seperti ini



Untuk persamaannya akan menjadi seperti di bawah ini kenapa untuk hasil yang Sin dikalikan dengan 30 karena jika tidak dikalikan maka hasilnya akan berada di range $0 < x < 1$ untuk memperjelas hasil grafiknya akan menjadi seperti ini



TASK 5

Kemudian untuk task 5 dengan persoalan seperti berikut

Buatlah tiga fungsi attribute garis, merupakan modifikasi dari linedda

dan bersenham pilih dengan ketentuan sbb:

- Tambahkan parameter jenis garis, gap/kerapatan dan warna pada fungsi linedda dan bersenham.
- Terdapat tiga jenis garis
- Dash-dash
- Point-dash-point
- Point-point
- Modifikasi hasil pertanyaan no 4 dengan fungsi garis yang telah dimodifikasi.

Untuk memerihatkannya dapat dengan sourcode di bawah ini dimana menggunakan perulangan for untuk memprint setiap dot dimana untuk hasila yang terdaapt garis ataupun dot menggunakan perulangan kemudian menggunakan putPixel untuk membuatnya kemudian menggunakan warna transparent untuk memberikan warna transparannya kemudian dengan mengatur persamaannya disetiap perulangannya

```
1 extends "res://Scene 1/Geometrik.gd"
2
3
4 # Declare member variables here. Examples:
5 # var a = 2
6 # var b = "text"
7
8 func linear1():
9     var x = -200
10    var warna = Color(0,0,0,0);
11    var warna2 = Color.yellow
12
13    while(x <500):
14        for i in 10:
15            var y = 2*x + 3
16            x+=0.5;
17            put_pixel(x+center_width, center_height-y, warna2)
18        for i in 10:
19            var y = 2*x + 3
20            x+=0.5
21            put_pixel(x+center_width, center_height-y, warna)
22
```

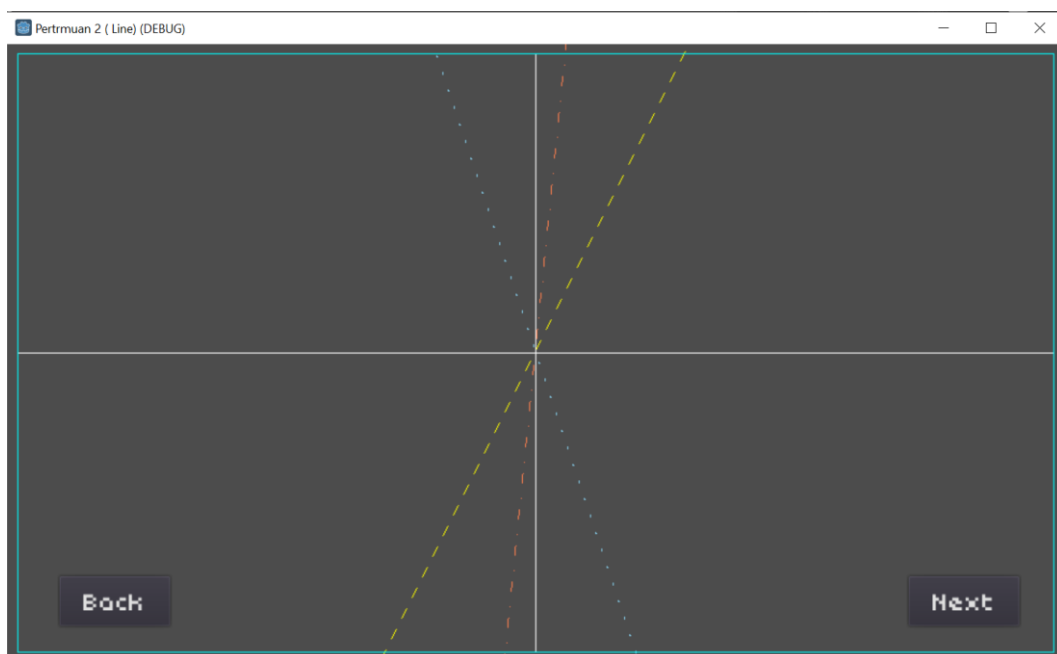
```
23 func linear2():
24     var x = -200
25     var warna = Color(0,0,0,0);
26     var warna2 = Color.skyblue
27
28     while(x <500):
29         for i in 5:
30             var y = -3*x + 2
31             x+=0.1;
32             put_pixel(x+center_width, center_height-y, warna2)
33         for i in 5:
34             var y = + -3*x + 2
35             x+=1;
36             put_pixel(x+center_width, center_height-y, warna)
37
38
39 func linear3():
40     var x = -200
41     var warna = Color(0,0,0,0);
42     var warna2 = Color.coral
43
```

```

44 > | while(x <1000):
45 > |     | for i in 1:
46 > |     |     | var y = 10*x + 4
47 > |     |     | x+=0.5;
48 > |     |     | put_pixel(x+center_width, center_height-y, warna2)
49 > |     | for i in 1:
50 > |     |     | var y = + 10*x + 4
51 > |     |     | x+=1;
52 > |     |     | put_pixel(x+center_width, center_height-y, warna)
53 > |     | for i in 10:
54 > |     |     | var y = 10*x + 4
55 > |     |     | x+=0.1;
56 > |     |     | put_pixel(x+center_width, center_height-y, warna2)
57 > |     | for i in 1:
58 > |     |     | var y = + 10*x + 4
59 > |     |     | x+=1;
60 > |     |     | put_pixel(x+center_width, center_height-y, warna)
61 > |     |
62
63 > func _on_Control_draw():
64 > |     linear1()
65 > |     linear2()
66 > |     linear3()
67 > |     kartesian()
68 > |     border()

```

Maka setelah di lihat dalam hasilnya akan menjadi seperti di bawah ini



Lesson Learn

Hal yang saya dapatkan setelah mempelajari materi ini saya sebelumnya hanya berfikir bahwa teori hanya sekedar teori saja tetapi dari mata kuliah dan proyek ini sangat berhubungan sekali jika dimana jika saya belum mengetahui teori ataupun teori saya belum matang maka hasil dari praktikum itu menyesuaikan dengan hasilnya. Ketika saya kuliah di sini karena banyaknya dengan praktikum kadang menyepelekan dari kuliah teori dan juga disini saya lebih memahami bagaimana cara untuk membentuk sebuah pixel dan juga sebuah line.

