**PRAKTIKUM 4**

1. **TUJUAN PRAKTIKUM.**

Setelah mengikuti praktikum ini, mahasiswa diharapkan dapat untuk :

1. Memahami pemrograman mode continuous

2. Memahami model animasi sederhana

1. **TEORI DASAR**

Struktur program dalam mode continuous adalah sebagai berikut :

1. Setiap program hanya mempunyai satu fungsi setup() dan draw().
2. Setup () akan dipanggil saat pertama kali program berjalan, kemudian secara berkala fungsi draw() dijalankan.
3. Jika perintah noLoop () dipanggil pada fungsi setup () maka fungsi draw() hanya akan dipanggil sekali saat program berjalan pertama kali.

Struktur Program Mode Contiuous <data global> Void setup()

{

}

pernyataan;

Void draw()

{

pernyataan;

}

1. PELAKSANAAN PRAKTIKUM

1. Membedakan penggunaan fungsi noLoop() dan dan tanpa noLoop()

Lakukan percobaan menggunakan listing berikut ini, abaikan arti dari perintah-perintah yang ada di dalam program tersebut, tetapi konsentrasikan perhatian terhadap apa yang dilihat saat program dijalankan.

Program 4.1

//sketch menggunakan noLoop()

void setup()

{

noLoop();

}

void draw ()

{

background (200);

int s = second ();

int m = minute();

int h = hour();

line(0,s,33,s);

line(33,m,66,m);

line(66,h,100,h);

}

Jalankan sketch diatas, perhatikan garis tidak berubah lokasi. Hentikan sketch tersebut dan kemudian letakkan tanda // di depan noLoop() pada fungsi setup(). Kemudian jalankan kembali sketch tersebut, perhatikan garis digambar garis tersebut berubah lokasinya, hal ini terjadi karena perintah noLoop() dan tanpa noLoop seperti pada program 4.2

**Program 4.2**

//sketch tanpa menggunakan noLoop()

void setup()

{

//noLoop();

}

void draw ()

{

background (200);

int s = second ();

int m = minute();

int h = hour();

line(0,s,33,s);

line(33,m,66,m);

line(66,h,100,h);

}

2. Membuat animasi garis

**Program 4.3**

void setup()

{

size ( 400,400); // layar output

background (255); // warna layar putih

}

void draw ()

{

int d;

d=second();

strokeWeight(4);

stroke(200,0,0);

line(10,30,10+5\*d,30); // garis merah nergerak ke kanan

}

3. Membuat animasi kotak

**Program 4.4**

void setup()

{

size ( 400,400); // layar output

background (255); // warna layar putih

}

void draw ()

{

int d;

d=second();

strokeWeight(4);

fill(200,0,200);

rect(10,10,50+5\*d,50+d\*5);

// kotak dari pojok kiri membesar ke kanan

}

4. Animasi Sederhana

**Program 4.5**

float x, y, z;

float size = 75;

void setup()

{

size(1000,500);

frameRate(30);

}

void draw()

{

background(25,25,112);

int d;

d=second();

noStroke();

//matahari

fill(0+50\*d,0,0);

ellipse(75,70,100,100);

//Jalan

fill(150,75,50);

rect(0,400,1000,100);

//gunung

strokeWeight(0);

fill(0,94,0);

arc(350,450,800,550,(180\*PI)/180,(360\*PI)/180);

fill(0,132,0);

arc(750,450,750,450,(180\*PI)/180,(360\*PI)/180);

//efek salju

fill(145);

arc(340,225,440,100,(180\*PI)/180,(0\*PI)/180);

arc(770,260,400,85,(180\*PI)/180,(0\*PI)/180);

ellipse(200,245,200,100);

ellipse(375,245,200,130);

ellipse(500,250,200,100);

ellipse(680,270,200,100);

ellipse(800,270,150,110);

ellipse(900,270,150,70);

//Rumah

fill(255,0,255);

triangle(120,350,90,375,150,375);

fill(20,100,200);

rect(90,375,60,30);

fill(0,100,0);

arc(120,405,30,50,(180\*PI)/180,(0\*PI)/180);

//

fill(0);

triangle(80,375,50,400,110,400);

fill(255);

rect(50,400,60,30);

fill(100);

arc(80,430,30,50,(180\*PI)/180,(0\*PI)/180);

//

fill(200,100,0);

triangle(160,375,130,400,190,400);

fill(200,0,0);

rect(130,400,60,30);

fill(0,200,200);

arc(160,430,30,50,(180\*PI)/180,(0\*PI)/180);

//

fill(100,100,100);

triangle(200,325,170,350,230,350);

fill(255,255,0);

rect(170,350,60,30);

fill(255,70,70);

arc(200,380,30,50,(180\*PI)/180,(0\*PI)/180);

//rumput

strokeWeight(3);

fill(34,139,34);

triangle(750,350,735,450,765,450);

triangle(780,350,765,450,795,450);

triangle(810,350,795,450,825,450);

triangle(840,350,825,450,855,450);

//salju

fill(255);

ellipse(50,5+10\*d,10,10);

ellipse(50+10\*d,5+10\*2\*d,10,10);

ellipse(50,5+10\*3\*d,10,10);

ellipse(100,0+10\*d,20,20);

ellipse(100,0+10\*d\*3,20,20);

ellipse(100+5\*d,0+10\*d\*2,20,20);

ellipse(200-10\*d,0+10\*3\*d,20,20);

ellipse(200,0+10\*d,20,20);

ellipse(275+10\*d,5+10\*d,15,15);

ellipse(275,0+10\*d,15,15);

ellipse(300,0+10\*d,20,20);

ellipse(350+10\*d,0+10\*2\*d,20,20);

ellipse(350,0+10\*d,20,20);

ellipse(350-10\*d,0+10\*d\*2,20,20);

ellipse(450-5\*d,0+10\*d,17.5,17.5);

ellipse(50,30+10\*d,10,10);

ellipse(500,5+10\*d,10,10);

ellipse(500-10\*d,5+10\*d\*3,10,10);

ellipse(500,5+10\*d\*2,10,10);

ellipse(600+10\*d,0+10\*d,20,20);

ellipse(650,0+10\*3\*d,20,20);

ellipse(675,0+10\*d,20,20);

ellipse(750+10\*d,5+10\*4\*d,15,15);

ellipse(800,0+10\*d,15,15);

ellipse(850,0+10\*d,20,20);

ellipse(850+10\*d,0+10\*d\*3,20,20);

ellipse(850-10\*d,0+10\*d\*2,20,20);

ellipse(875,0+10\*2\*d\*3,20,20);

ellipse(875-5\*d,0+10\*2\*d,20,20);

ellipse(900-5\*d,0+10\*d,17.5,17.5);

ellipse(900+5\*d,0+10\*d\*1,20,20);

ellipse(900,0+10\*d\*3,20,20);

//awan

y = y + 0.8;

translate(y-200, height/2-2\*size/2);

fill(95,158,160);

ellipse(450,-50,70,45);

ellipse(500,-50,85,65);

ellipse(540,-50,70,35);

ellipse(700,-100,80,50);

ellipse(750,-100,95,75);

ellipse(780,-100,80,55);

ellipse(815,-100,75,40);

ellipse(250,-100,80,35);

ellipse(300,-100,80,45);

//mobil

fill(106,90,205);

rect(100,250,100,50);

fill(72,61,139);

rect(200,265,40,35);

fill(255);

rect(210,272,20,20);

fill(65);

ellipse(130,300,30,30);

ellipse(200,300,30,30);

x=x+0.5;

translate(x, height/2-3\*size/2);

fill(255,20,147);

rect(200,100,100,50);

fill(255,105,180);

rect(300,115,40,35);

fill(255);

rect(310,122,20,20);

fill(65);

ellipse(230,150,30,30);

ellipse(300,150,30,30);

translate(y,size);

fill(0,255,0);

rect(200,50,100,50);

fill(124,202,0);

rect(300,65,40,35);

fill(255);

rect(310,72,20,20);

fill(65);

ellipse(230,100,30,30);

ellipse(300,100,30,30);

}

1. **LAPORAN PRAKTIKUM**

|  |  |
| --- | --- |
| **Program** | **Hasil Program** |
| Program 4.1 | Hasil Program 4.1 |
| Program 4.2 | Hasil Program 4.2 |
| Program 4.3 | Hasil Program 4.3 |
| Program 4.4 | Hasil Program 4.4 |
| Program 4.5 | Program 4.5 |