1.Soal Algoritma Garis:

Algoritma Bressenham untuk membentuk garis adalah sebagai berikut :

1.Tentukan dua titik yang akan dihubungkan dalam pembentukan garis.
2. Tentukan salah satu titik disebelah kiri sebagai titik awal, yaitu (x0, y0) dan titik lainnya sebagai titik akhir (xe, ye).
3. Hitung dx, dy, 2 dx dan 2 dy – 2 dx.
4. Hitung parameter : p0 = 2 dy – dx.
5. Untuk setiap xk sepanjang jalur garis, diawali dengan k = 0, dan

Jika pk < 0, maka titik selanjutnya adalah (xk +1, yk) dan pk+1 = pk + 2 dy
Jika tidak, maka titik selanjutnya adalah (xk +1, yk + 1) dan pk+1 = pk + 2 dy – 2 dx.

6. Ulangi langkah 5. untuk menentukan posisi

Berikut ini diberikan contoh algoritma garis Bressenham dalam pembentukan suatu garis yang menghubungkan titik (10, 10) dan (17, 16),

Pertama-tama ditentukan bahwa titik (10, 10) berada di sebelah kiri dan ini merupakan titik awal, sedangkan titik (17, 16) merupakan titik akhir.

Posisi pixel yang membentuk garis dapat ditentukan dengan melalui perhitungan sebagai berikut :

dx = xe – x0 dan dy = ye – y0

parameter p0 = 2 dy – dx.

k	p_k	$(x_k +1, y_k + 1$
	(10, 10)	
0	3	(11, 11)
1	1	(12, 12)
3	-1	(13, 13)
4	11	(14, 13)
5	9	(15, 14)
6	7	(16, 15)
7	5	(17, 16)

Untuk menggambarkan algoritma DDA dalam pembentukan suatu garis yang menghubungkan titik (10,10) dan (17,16), pertama-tama ditentukan dx dan dy, kemudian dicari step untuk mendapatkan x_increment dan y_increment.

Δx = x1 — x 0 = 17–10 = 7
Δy = y1 — y0 = 16 -10 = 6

selanjutnya hitung dan bandingkan nilai absolutnya.

|Δx| = 7
|Δy| = 6

karena |Δx| > |Δy|, maka step = |Δx| = 7, maka diperoleh :

x_inc = 7/7= 1
y_inc = 6/7 = 0,86 .

k	x	y	round(x),round(y)
			(10,10)
0	11	10,86	(11,11)
1	12	11,72	(12,12)
2	13	12,58	(13,13)
3	14	13,44	(14,13)
4	15	14,3	(15,14)
5	16	15,16	(16,15)
6	17	16,02	(17,16)

2.Soal Transformasi:

Translasi dengan vektor (2, 3) pada objek dengan titik-titik (1, 1), (2, 3), dan (4, 2):

(1, 1) + (2, 3) = (3, 4)
(2, 3) + (2, 3) = (4, 6)
(4, 2) + (2, 3) = (6, 5)

Rotasi objek dengan sudut 45 derajat searah jarum jam menggunakan titik pusat rotasi (0, 0):

Titik-titik setelah rotasi adalah:

(1, 1) rotasi = ($\sqrt{2}$, $\sqrt{2}$)
(2, 3) rotasi = (1.1, 4.24)
(4, 2) rotasi = (2.12, 2.83)

3.Soal Clipping:

Menggunakan algoritma Cohen-Sutherland pada garis dengan titik awal (10, 20) dan titik akhir (30, 40) untuk mengklipnya ke dalam window dengan batas (15, 25) dan (25, 35):

Langkah 1: Tentukan kode pengkodean (outcode) untuk setiap titik. Untuk titik awal (10, 20), kode outcode adalah 1001, dan untuk titik akhir (30, 40), kode outcode adalah 0110.

Langkah 2: Lakukan pemeriksaan logika bitwise antara kedua kode outcode. Jika hasilnya bukan 0000, maka garis harus dipotong.

Langkah 3: Lakukan iterasi untuk mengklip garis:
Iterasi 1: Cek kode outcode titik awal (10, 20). Kode outcode pada posisi kiri teratas adalah 1, yang berarti titik berada di luar batas kiri window. Potong garis dengan garis potong dengan batas kiri window pada (15, 20).

Iterasi 2: Cek kode outcode titik akhir (30, 40). Kode outcode pada posisi kanan bawah adalah 1, yang berarti titik berada di luar batas kanan window. Potong garis dengan garis potong dengan batas kanan window pada (25, 40).

Hasil klipping adalah garis baru dengan titik awal (15, 20) dan titik akhir (25, 40).

Algoritma Garis:
a. Algoritma DDA (Digital Differential Analyzer) //based on elon

Soal:
Implementasikan algoritma DDA untuk menggambar garis dari titik (2, 2) ke titik (8, 6). Jika nilai delta x dan delta y adalah 6 dan 4, tentukan koordinat titik-titik antara kedua titik tersebut.

Jawaban:
Berikut adalah langkah-langkah pengerjaan dengan algoritma DDA:

Tentukan delta x dan delta y:
delta x = x2 - x1 = 8 - 2 = 6
delta y = y2 - y1 = 6 - 2 = 4
Tentukan langkah yang lebih panjang (step):
Jika |delta x| > |delta y|, maka step = |delta x| = 6
Jika |delta y| > |delta x|, maka step = |delta y| = 4
Jika |delta x| = |delta y|, maka step = |delta x| = 6 (atau bisa juga menggunakan |delta y|)
Hitung increment untuk x dan y:
increment x = delta x / step = 6 / 6 = 1
increment y = delta y / step = 4 / 6 ≈ 0.67
Mulai dari titik awal (x1, y1) = (2, 2) dan lakukan perulangan sebanyak step kali:
Untuk setiap perulangan, tambahkan increment x dan increment y ke x dan y saat ini.
Hasilkan koordinat titik-titik antara kedua titik.
Berhenti ketika mencapai titik akhir (x2, y2) = (8, 6).
Hasil:
Koordinat titik-titik antara (2, 2) dan (8, 6) adalah sebagai berikut:
(2, 2), (3, 3), (4, 4), (5, 5), (6, 5), (7, 6), (8, 6)

b. Algoritma Bresenham //based on elon

Soal:
Implementasikan algoritma Bresenham untuk menggambar garis dari titik (3, 2) ke titik (9, 7). Tentukan koordinat titik-titik antara kedua titik tersebut.

Jawaban:
Berikut adalah langkah-langkah pengerjaan dengan algoritma Bresenham:

Tentukan delta x dan delta y:
delta x = x2 - x1 = 9 - 3 = 6
delta y = y2 - y1 = 7 - 2 = 5
Tentukan nilai awal error = 0.
Hitung increment untuk x dan y:
Jika |delta x| > |delta y|, maka increment x = sign(delta x) = sign(6) = 1
dan increment y = delta y / delta x = 5 / 6 ≈ 0.83
Jika |delta y| > |delta x|, maka increment x = delta x / delta y = 6 / 5 = 1.2

dan increment y = sign(delta x) = sign(5) = 1

Jika |delta x| = |delta y|, maka increment x = sign(delta x) = sign(6) = 1
dan increment y = sign(delta y) = sign(5) = 1
Mulai dari titik awal (x1, y1) = (3, 2) dan lakukan perulangan sebanyak langkah terpanjang (|delta x| atau |delta y|) kali:
Untuk setiap perulangan, tambahkan increment x dan increment y ke x dan y saat ini.
Jika error > 0.5, tambahkan increment y ke y saat ini dan kurangi error dengan 1.
Jika error ≤ 0.5, lanjutkan tanpa mengubah y dan kurangi error dengan 0.
Hasilkan koordinat titik-titik antara kedua titik.
Berhenti ketika mencapai titik akhir (x2, y2) = (9, 7).
Hasil:
Koordinat titik-titik antara (3, 2) dan (9, 7) adalah sebagai berikut:
(3, 2), (4, 3), (5, 3), (6, 4), (7, 5), (8, 6), (9, 7)

Transformasi:
a. Translasi (pemindahan objek)

Soal:
Diberikan titik-titik objek A dengan koordinat (2, 4), (3, 6), dan (5, 8). Lakukan translasi objek A sebesar tx = 3 dan ty = -2. Tentukan koordinat titik-titik objek A setelah translasi.

Jawaban:
Langkah-langkah untuk melakukan translasi:

Tentukan nilai tx = 3 dan ty = -2 sebagai pergeseran dalam sumbu x dan y.
Untuk setiap titik (x, y) dalam objek A, lakukan penjumlahan tx dan ty terhadap koordinat titik tersebut.
Hasilkan koordinat titik-titik objek A setelah translasi.
Hasil:
Koordinat titik-titik objek A setelah translasi adalah:
(2 + 3, 4 + (-2)) = (5, 2)
(3 + 3, 6 + (-2)) = (6, 4)
(5 + 3, 8 + (-2)) = (8, 6)

b. Skalasi (perubahan ukuran objek)

Soal:
Diberikan titik-titik objek B dengan koordinat (2, 3), (4, 6), dan (6, 9). Lakukan skalasi objek B dengan faktor skala sx = 2 dan sy = 1.5. Tentukan koordinat titik-titik objek B setelah skalasi.

Jawaban:
Langkah-langkah untuk melakukan skalasi:

Tentukan nilai sx = 2 dan sy = 1.5 sebagai faktor skala dalam sumbu x dan y.
Untuk setiap titik (x, y) dalam objek B, lakukan perkalian sx dan sy terhadap koordinat titik tersebut.

Hasilkan koordinat titik-titik objek B setelah skalasi.
Hasil:
Koordinat titik-titik objek B setelah skalasi adalah:
(2 * 2, 3 * 1.5) = (4, 4.5)
(4 * 2, 6 * 1.5) = (8, 9)
(6 * 2, 9 * 1.5) = (12, 13.5)

c. Rotasi (memutar objek)

Soal:
Diberikan titik-titik objek C dengan koordinat (1, 1), (2, 2), dan (3, 3). Lakukan rotasi objek C sebesar sudut theta = 45 derajat terhadap pusat rotasi (0, 0). Tentukan koordinat titik-titik objek C setelah rotasi.

Jawaban:
Langkah-langkah untuk melakukan rotasi:

Tentukan sudut theta = 45 derajat sebagai sudut rotasi.
Untuk setiap titik (x, y) dalam objek C, lakukan rotasi dengan menggunakan rumus:
x' = x * cos(theta) - y * sin(theta)
y' = x * sin(theta) + y * cos(theta)
Hasilkan koordinat titik-titik objek C setelah rotasi.
Hasil:
Koordinat titik-titik objek C setelah rotasi adalah:
(1 * cos(45) - 1 * sin(45), 1 * sin(45) + 1 * cos(45)) ≈ (0, 1.41)
(2 * cos(45) - 2 * sin(45), 2 * sin(45) + 2 * cos(45)) ≈ (0, 2.83)
(3 * cos(45) - 3 * sin(45), 3 * sin(45) + 3 * cos(45)) ≈ (0, 4.24)

d. Refleksi (memantulkan objek)

Soal:
Diberikan titik-titik objek D dengan koordinat (1, 2), (3, 4), dan (5, 6). Lakukan refleksi objek D terhadap sumbu x. Tentukan koordinat titik-titik objek D setelah refleksi.

Jawaban:
Langkah-langkah untuk melakukan refleksi:

Lakukan perubahan tanda terhadap komponen y (y negatif) pada setiap titik (x, y) dalam objek D.
Hasilkan koordinat titik-titik objek D setelah refleksi.
Hasil:
Koordinat titik-titik objek D setelah refleksi terhadap sumbu x adalah:
(1, -2)
(3, -4)
(5, -6)

Clipping:
a. Cohen-Sutherland (clipping garis dengan window)

Soal:
Terapkan algoritma Cohen-Sutherland pada garis yang didefinisikan oleh titik (2, 3) dan (8, 6) dengan window yang didefinisikan oleh titik (4, 4) dan (6, 8). Tentukan koordinat titik awal dan akhir garis setelah proses clipping.

Jawaban:
Langkah-langkah untuk melakukan clipping dengan algoritma Cohen-Sutherland:

Tentukan kode kategori (outcode) untuk kedua titik garis dan window menggunakan aturan berikut:

Bit pertama (kiri): 1 jika titik berada di sebelah kiri window, 0 jika tidak.
Bit kedua (kanan): 1 jika titik berada di sebelah kanan window, 0 jika tidak.
Bit ketiga (bawah): 1 jika titik berada di bawah window, 0 jika tidak.
Bit keempat (atas): 1 jika titik berada di atas window, 0 jika tidak.
Lakukan perulangan untuk setiap titik garis:
Tentukan outcode untuk titik awal dan titik akhir garis.
Jika kedua outcode sama dan bernilai 0000 (titik sepenuhnya berada dalam window), maka garis tidak perlu dipotong.
Jika hasil operasi bitwise AND dari kedua outcode tidak sama dengan 0000, maka garis sepenuhnya berada di luar window dan tidak perlu ditampilkan.
Jika hasil operasi bitwise AND dari kedua outcode sama dengan 0000, maka garis memotong window. Selanjutnya, lakukan proses pemotongan menggunakan aturan berikut:
Tentukan titik potong menggunakan algoritma garis yang sesuai (misalnya, DDA atau Bresenham).
Tentukan titik potong baru untuk garis.
Setelah proses pemotongan selesai, tentukan koordinat titik awal dan akhir garis yang telah dipotong.
Hasil:
Koordinat titik awal dan akhir garis setelah proses pemotongan dengan algoritma Cohen-Sutherland adalah:
Titik awal: (4, 4)
Titik akhir: (6, 6)

b. Liang-Barsky (clipping garis dengan polygon)

Soal:
Terapkan algoritma Liang-Barsky pada garis yang didefinisikan oleh titik (2, 4) dan (8, 6) dengan polygon yang didefinisikan oleh batas koordinat (4, 5), (6, 5), (6, 7), dan (4, 7). Tentukan koordinat titik awal dan akhir garis setelah proses clipping.

Jawaban:
Langkah-langkah untuk melakukan clipping dengan algoritma Liang-Barsky:

Tentukan parameter p dan q untuk setiap sisi polygon berdasarkan batas koordinat.

Lakukan perulangan untuk setiap sisi polygon:
Tentukan nilai r1 dan r2 menggunakan rumus r = q / (-p).
Jika p = 0 dan q < 0, maka garis berada di luar polygon dan tidak perlu ditampilkan.
Jika p < 0, perbarui nilai r1 dengan membandingkannya dengan r1 sebelumnya. Jika p > 0, perbarui nilai r2 dengan membandingkannya dengan r2 sebelumnya.
Jika r1 > r2, maka garis berada di luar polygon dan tidak perlu ditampilkan.
Jika r1 <= r2, lakukan pemotongan garis menggunakan rumus-rumus berikut:
Tentukan nilai u1 dan u2 menggunakan rumus u = r.
Tentukan titik potong baru untuk garis.
Setelah proses pemotongan selesai, tentukan koordinat titik awal dan akhir garis yang telah dipotong.
Hasil:
Koordinat titik awal dan akhir garis setelah proses pemotongan dengan algoritma Liang-Barsky adalah:
Titik awal: (4.33, 5)
Titik akhir: (6, 5.67)

Catatan:
Pada kedua algoritma clipping di atas, proses pemotongan garis dilakukan berdasarkan batas window atau polygon yang diberikan. Garis yang sepenuhnya berada di luar batas akan diabaikan, sedangkan garis yang memotong batas akan dipotong dan titik potong baru ditentukan.

Pengantar Grafik Komputer:

grafika komputer adalah serangkaian langkah atau prosedur yang digunakan untuk menghasilkan gambar atau objek visual dalam komputer. Tujuannya adalah untuk mengatur tata letak, pencahayaan, warna, dan efek visual lainnya.
Contoh algoritma dasar dalam grafika komputer meliputi algoritma Bresenham untuk menggambar garis, algoritma Flood Fill untuk pewarnaan, dan algoritma Transformasi Geometri seperti translasi, rotasi, dan penskalaan.
Algoritma sangat penting dalam grafika komputer karena membantu dalam mengoptimalkan proses penggambaran, meningkatkan efisiensi, dan menghasilkan gambar yang lebih realistis atau kompleks.
OpenGL:

OpenGL (Open Graphics Library) adalah API (Application Programming Interface) yang digunakan untuk mengembangkan aplikasi grafika komputer 2D dan 3D. Tujuan utama OpenGL adalah menyediakan antarmuka umum yang dapat digunakan untuk berinteraksi dengan perangkat keras grafis.
Fungsi utama OpenGL meliputi rendering 2D dan 3D, manajemen tekstur, pencahayaan, transformasi geometri, dan pengolahan gambar. Dengan menggunakan OpenGL, pengembang dapat membuat aplikasi grafika yang kaya dan interaktif dengan mudah.
Contoh penggunaan OpenGL termasuk pengembangan permainan komputer, simulasi fisika, visualisasi data, dan desain grafis.
Kurva:

Dalam grafika komputer, kurva adalah bentuk geometris yang dihasilkan oleh serangkaian titik kontrol yang mengendalikan jalur atau bentuk yang diinginkan. Kurva digunakan untuk menggambarkan objek yang lebih halus, seperti kurva Bezier, kurva B-spline, kurva lingkaran, atau kurva Hermite.
Contoh kurva Bezier adalah kurva yang digunakan untuk menghasilkan garis lengkung atau bentuk yang lebih kompleks. Persamaan matematika yang digunakan untuk menggambarkan kurva Bezier adalah $B(t) = (1-t)^3 * P0 + 3(1-t)^2 * t * P1 + 3(1-t) * t^2 * P2 + t^3 * P3$, di mana P0, P1, P2, dan P3 adalah titik kontrol.

Kurva B-spline adalah kurva yang digunakan untuk menghasilkan bentuk yang lebih fleksibel dan halus. Persamaan matematika yang digunakan untuk menggambarkan kurva B-spline melibatkan fungsi basis dan matriks kontrol.

Algo DDA

- $(x_1, y_1), (x_2, y_2)$

- $\Delta x = x_2 - x_1$
$\Delta y = y_2 - y_1$ } $\Delta y > \Delta x \rightarrow step = |\Delta y| \; || \; \Delta x \rightarrow step$

x-inc = Δx/step
y-inc = Δy/step } $\rightarrow$ (x+x-inc, y+y-inc)

Bersenhem

$0<m<1$ $\quad (x_1,y_1),(x_2,y_2) \rightarrow m = \left(\frac{y_2-y_1}{x_2-x_1}\right)$

$dx = x_2 - x_1$
$dy = y_2 - y_1$
$d_1 = 2 \cdot dy$
$d_2 = 2 \cdot (dx - dy)$
$p = d_1 \cdot dx$

Jika $p >= 0$
$p = p - d_2$
$y = y+1$
$x = x+1$

Jika $p < 0$
$p = p + d_1$
$y = y$
$x = x+1$

$m > 1$

$dx = x_2 - x_1$
$dy = y_2 - y_1$
$d_1 = 2 \cdot dx$
$d_2 = 2 \cdot (dx - dy)$
$p = d_1 - dy$

Jika $p >= 0$
$p = p + d_2$
$x = x+1$
$y = y+1$

Jika $p < 0$
$p = p + d_1$
$x = x$
$y = y+1$

Midle point

$\Delta x = x_2 - x_1$
$\Delta y = y_2 - y_1$
$D_{init} = 2\Delta y - \Delta x$

$\Delta D = 2(\Delta y - \Delta x)$

Jika $D_{init} < 0$
$x = x+1$
$y = y$
$D_{new} = D_{init} + 2\Delta y$

Jika $D_{init} >= 0$
$x = x+1$
$y = y+1$
$D_{new} = D_{init} + \Delta D$

Dot product

Panjang vektor $r \rightarrow |r|$, nilainya $\sqrt{r \cdot r}$
Sudut antara 2 vektor : $\theta = \cos^{-1}\left(\frac{r \cdot s}{|r||s|}\right)$

$|s| = |r| \cos\theta = |r| \left(\frac{r \cdot u}{|r||u|}\right) = r \cdot u$

Translasi
$\rightarrow (x,y) \rightarrow$ transalsi/geser
$Q(x,y) = P(x,y) + Tr$
$= P(x+Tr, y+Tr)$

Scaling
$\rightarrow Q(x,y) = A \cdot S$
$= A(x,y) \cdot S$
$= A(x \cdot S, y \cdot S)$

Rotasi
(0,0)
$\cos\theta = \frac{x}{r}$, $\sin\theta = \frac{y}{r}$

$x = r \times \cos\theta$ $\quad y = r \times \sin\theta$

$x' = x \cdot \cos\theta - y \sin\theta$
$y' = x \cdot \sin\theta + y \cos\theta$

Sembarang
Translasi $\rightarrow (-x, -y)$
rotasi / scaling
Translasi $\rightarrow (x, y)$

clipp window → persegi empat dgn posisi standar ($P(x, y)$) $xw_{min} \le x \le xw_{max}$ | $yw_{min} \le y \le yw_{max}$

Garis → inside-outside tes dgn memeriksa endpoint dari garis. garis punya kedua endpoint dlm batas clipping → disimpan

bit 1 = t → atas window ($y_{max} - y$)
bit 2 = t → bawah window ($y - y_{min}$)
Bit 3 = t → kanan window ($x_{max} - x$)
bit 4 = t → kiri window ($x - x_{min}$)

Region code:

3	2	1	0
T	B	R	L

T = 1 Jika ujung garis diatas area gambar
B = 1 —— " —— dibawah —— " ——
L = 1 —— " —— dikiri —— " ——
R = 1 —— " —— dikanan —— " ——

$C_0 \wedge C_1$ → trivial reject
kedua ujung ada dalam suatu baris / kolom outside

$T = (xp1, y_{max})$, $B = (xp2, y_{min})$, $R = (x_{max}, yp1)$
$L = (x_{min}, yp2)$

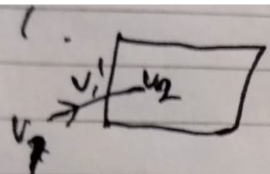
$$xp_1 = x_1 + \frac{y_{max} - y_1}{m}, \quad xp_2 = x_1 + \frac{y_{min} - y_1}{m}$$

$$yp_1 = y_1 + m \times (x_{max} - x_1), \quad yp_2 = y_1 + m \times (x_{min} - x_1)$$

A	B	
0	0	0
0	1	0
1	0	0
1	1	1

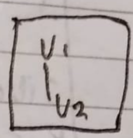
Potong
T = y_{max} = (xp_1, y_{max})
B = y_{min} = ($xp2$, y_{min}
R = x_{max} = (x_{max}, yp_1)
L = x_{min} = (x_{min}, $yp2$)

$\begin{pmatrix} \cos\theta & -s \\ \sin\theta & c \end{pmatrix}$

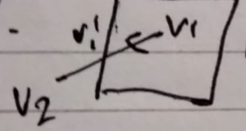
1.


out-in

save: V_2, V_1'

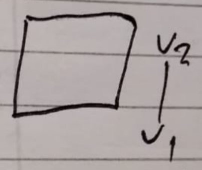
2.


save = V_2

3.


in-out

save = V_1'

4.


save = nothing

A	30	45	60	90	120	135	180
sin	0	$1/2$	$\sqrt{3}/2$	1	$\sqrt{3}/2$	$1/\sqrt{2}$	0
cos	1	$1/2$	$1/2$	0	$-1/2$	$-1/\sqrt{2}$	-1
Tan	0	1	$\sqrt{3}$	UD	$-\sqrt{3}$	-1	0