**UTS 2022**

**Nama : Dion Sandy Ara Tambunan  
NIM : 20051397056  
Prodi : D4 Manajemen Informatika  
Kelas : 2020B**

**1. Dalam computer science terdapat beberapa sub bidang yang berhubungan dengan grafis meliputi computer graphics, Image processing, computer vision dan pattern recognition. Jelaskan perbedaannya.**Grafika komputer (Inggris: Computer graphics) adalah bagian dari ilmu komputer yang berkaitan dengan pembuatan dan manipulasi gambar (visual) secara digital. Bentuk sederhana dari grafika komputer adalah grafika komputer 2D yang kemudian berkembang menjadi grafika komputer 3D, pemrosesan citra (image processing), dan pengenalan pola (pattern recognition). Grafika komputer sering dikenal juga dengan istilah visualisasi data.

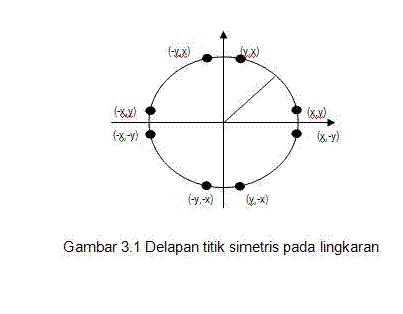
Computer Vision adalah ilmu dan teknologi mesin yang melihat, di mana mesin mampu mengekstrak informasi dari gambar yang diperlukan untuk menyelesaikan tugas tertentu. Sebagai suatu disiplinilmu, visi komputer berkaitan dengan teori di balik sistem buatan bahwa ekstrak informasi dari gambar. Data gambar dapat mengambil banyak bentuk, seperti urutan video, pandangan dari beberapa kamera, atau data multi-dimensi dari scanner medis.

Pengolahan Citra (Image Processing) merupakan proses memperbaiki kualitas citra agar mudah diinterpretasi oleh manusia atau komputer. Teknik pengolahan citra dilakukan dengan mentrasformasikan citra menjadi citra lain, misalnya: pemampatan citra (image compression).

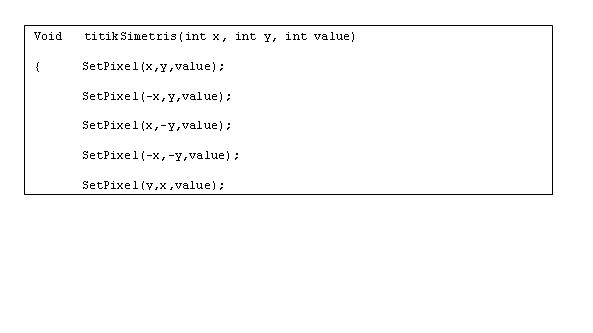
Pattern recognition adalah proses otomatis untuk menemukan keteraturan di dalam data melalui penggunaan algoritma komputer dengan tujuan untuk mengambil tindakan tindakan seperti mengklasifikasikan data kedalam kategori kategori berbeda.

**2. Jelaskan algorithma pembentukan garis Brassenham disertai kekurangan dan kelebihannya.**Algoritma ini digunakan untuk pembentukan garis menggunakan bilangan integer, sehingga tidak diperlukan adanya proses pembulatan bilangan pada setiap iterasinya. Algoritma Bresenham ini juga dikenal dengan istilah Midpoint Line Algorithm (teknik ini juga digunakan pada penggambaran lingkaran).  
  
\*Kelebihan :  
1) Algoritma ini lebih akurat dan digunakan pengurangan & penambahan untuk menghitung nilai piksel saat menggambar garis.  
2) Keakuratan algoritma Bresenham dapat diandalkan saat menggambar kurva & lingkaran

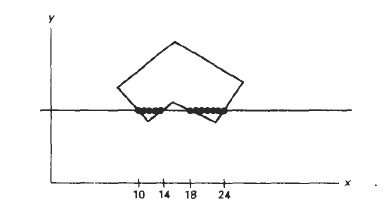
**3. Jelaskan tentang algorithma pembentukan lingkaran menggunakan 8 Titik Simetris.**Proses pembentukan lingkaran dengan algoritma ini dapat dilakukan dengan menentukan suatu titik awal. Bila titik awal pada lingkaran (x,y) maka terdapat tiga posisi lain, sehingga dapat diperoleh delapan titik. Dengan demikian sebenarnya hanya diperlukan untuk menghitung segmen 450 dalam menentukan lingkaran selengkapnya. Dengan titik pusat lingkaran yang tertentu, delapan titik simetris dapat ditampilkan seperti pada gambar berikut :

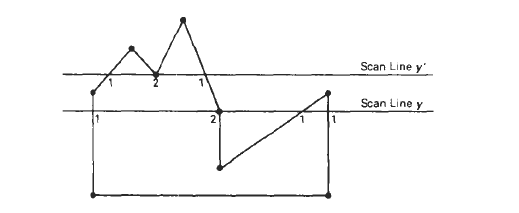


Persamaan untuk menentukan koordinat titik-titik yang terletak pada sebuah piksel dengan menentukan sebuah besarnya sudut terhadap sumbu X. Memang bisa menggambar lingkaran dengan seperti ini tapi memerlukan ketelitian yang cukup tinggi. Dalam pembuatan lingkaran dapat dilakukan dengan menentukan suatu titik awal. Dalam lingkaran terdapat empat posisi, sehingga dapat diperoleh delapan titik. Dari kedelapan titik simetris ini masing-masing akan dihubungkan untuk membentuk suatu lingkaran. Procedure pembentukan lingkaran dengan menggunakan 8 titik simetris adalah sebagai berikut :

****

**4. Jelaskan tentang algorithma Fill-area menggunakan Scan Line.**pengisian area dilakukan menurut arah scan line (garis scan) yang melintasi polygon kemudian posisi yang berhubungan antara sepasang titik tertentu diberi warna



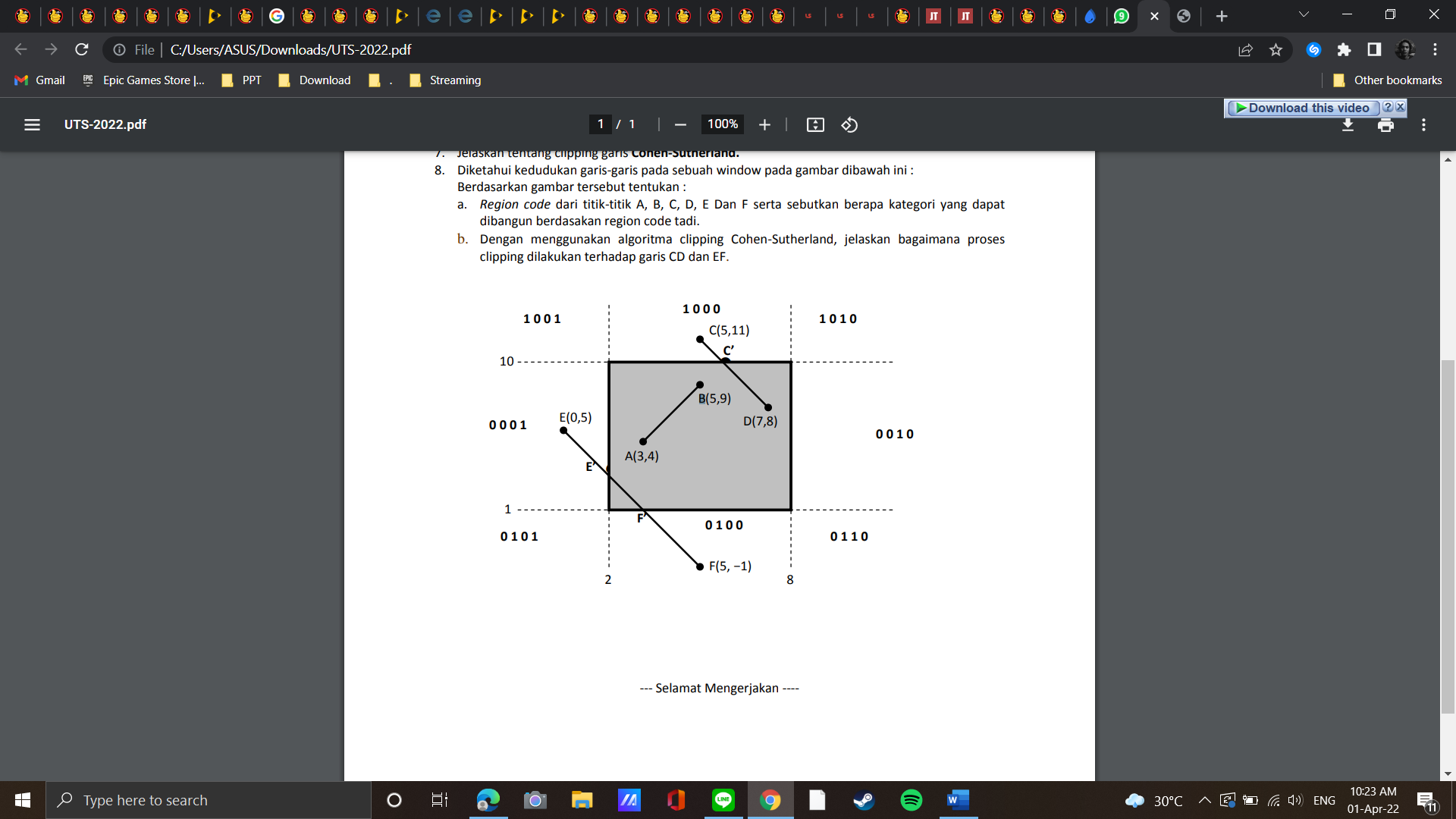


**5. Jelaskan perbedaan Boundary Fill dan Flood Fill  
Algoritma Boundary Fill** Metode ini dapat digunakan apabila titik dalam suatu bangun geometri sudah diketahui, algoritma boundary fill memerlukan titik koordinat (x,y), pola isi, dan pola batas

Sedangkan **Algoritma Flood fill**, metode ini dimulai dengan satu titik koordinat dalam satu bangun geometri, selanjutnya mendefinisikan seluruh bagian yg akan difill dengan warna yang sama. Apabila bagian yg akan di fill terdiri dari beberapa warna maka yg pertama harus dilakukan adalah membuat nilai pixel baru sehingga semua bagian yg akan difill mempunyai warna pixel yang sama

**7. Jelaskan tentang clipping garis Cohen-Sutherland**  
Salah satu algoritma **clipping** diusulkan oleh Danny **Cohen** and Ivan **Sutherland** pada tahun 1967 yaitu algoritma **Cohen**-**Sutherland**. Algoritma ini digunakan untuk menentukan **apakah** terdapat potongan **garis** yang digambar di dalam jendela dan sebaliknya akan menghilangkan potongan **garis** yang berada di luar jendela.

**8. Diketahui kedudukan garis-garis pada sebuah window pada gambar dibawah ini : Berdasarkan gambar tersebut tentukan : a. Region code dari titik-titik A, B, C, D, E Dan F serta sebutkan berapa kategori yang dapat dibangun berdasakan region code tadi. b. Dengan menggunakan algoritma clipping Cohen-Sutherland, jelaskan bagaimana proses clipping dilakukan terhadap garis CD dan EF.**



Jawab:

8

1

10

A(3,4)

B(5,9)

C(5,11)

D(7,8)

E(0,5)

F(5, −1)

G(0, 3)

I(10,8)

J(9,2)

**Window**

**1**

**0 0**

**0**

**0**

**1 0**

**0**

**0**

**0 1**

**0**

**0**

**0 0**

**1**

**0 0**

**1**

**1**

**0 1**

**1**

**0**

**0**

**1 0**

**1**

**1 1**

**0**

**0**

**C’**

**E**

**’**

**F**

**’**

**H**

**’**

**G**

**’**

•

•

•

•

•

Jawab:

A)

Titik

Region code

Kategori Titik

A(3,4)

0

0 0

0

visible

B(5,9)

0

0

0 0

visible

**’**

|  |  |  |
| --- | --- | --- |
| C(5,11) | 1 0 0 0 | invisible |
| D(7,8) | 0 0 0 0 | visible |
| E(0,5) | 0 0 0 1 | invisible |
| F(5,-1) | 0 1 0 0 | invisible |
| G(0,3) | 0 0 0 1 | invisible |
| H(3,-1) | 0 1 0 0 | invisible |
| I(10,8) | 0 0 1 0 | invisible |
| J(9,2) | 0 0 1 0 | invisible |

Kategori I : Garis AB visible, karena region code kedua ujungnya 0000

Kategori II : Garis I J invisible karena, region code I = 0010 , J = 0010 dan

0 0 1 0 AND 0 0 1 0 = 0 0 1 0 ≠ 0 0 0 0 Kategori III :

Garis CD *candidates for clipping*, karena 1 0 0 0 AND 0 0 0 0 = 0 0 0 0

Garis EF *candidates for clipping*, karena 0 0 0 1 AND 0 1 0 0 = 0 0 0 0

Garis GH *candidates for clipping*, karena 0 0 0 1 AND 0 1 0 0 = 0 0 0 0

B) Proses Clipping

### Clipping garis CD

Garis CD melewati titik C (5,11) region code 1000 (atas window) dan tittik D(7,8) region code 0000 (dalam window).

Gradien garis CD :

*m*

2

3

5

7

11

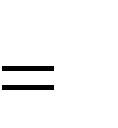
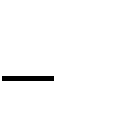
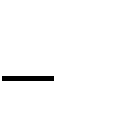
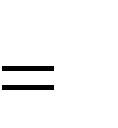
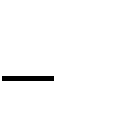
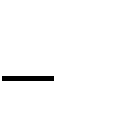
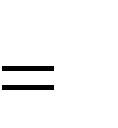
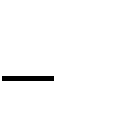
8

1

2

1

2



*x*

*x*

*y*

*y*

Titik potong C’ antara garis CD dengan batas atas window *y*max = 10 adalah

*x*

67

,

5

2

/

3

11

10

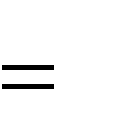
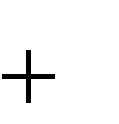
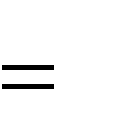
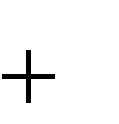
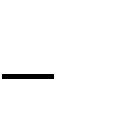
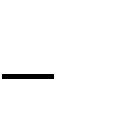
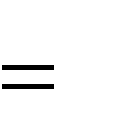
5

)

(

1

1



*x*

*m*

*y*

*y*

*x*

*x*

*batas*

Titik potong C’ (5,67 , 10) region code = 0000

Clipp garis CC’ dan gambar garis C’D, karena garis C’D region code kedua ujungnya 0000

### Clipping garis EF

Garis EF melewati titik E (0, 5) region code 0001 (kiri window) dan titik F(5, −1) region code 0100 (bawah window).

Gradien garis EF *m*

5

6

0

5

5

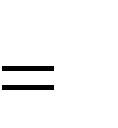
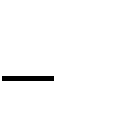
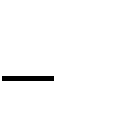
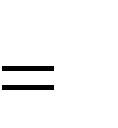
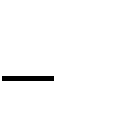
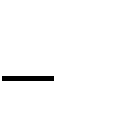
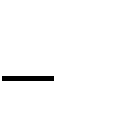
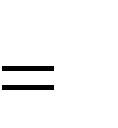
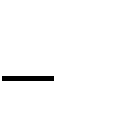
1

1

2

1

2



*x*

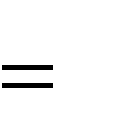
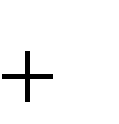
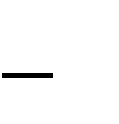
*x*

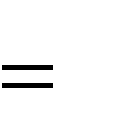
*y*

*y*

Titik potong E’ antara garis EF dengan batas kiri window *x*min = 2 adalah

*y y*1 *m*(*xbatas x*1)



*y* *y*2,6

)

0

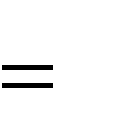
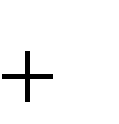
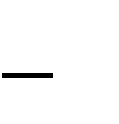
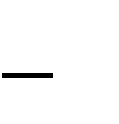
2

(

5

6

5



Titik potong E’(2, 2,6) region code = 0000

Titik potong F’ antara garis EF dengan batas bawah window *y*min = 1 adalah

*x*

67

,

2

3

8

2

/

3

5

1

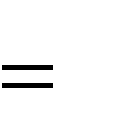
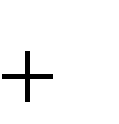
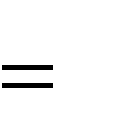
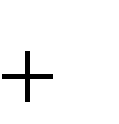
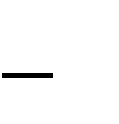
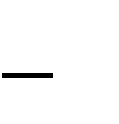
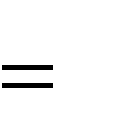
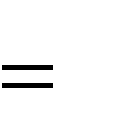
0

)

(

1

1



*x*

*m*

*y*

*y*

*x*

*x*

*batas*

Titik potong F’ (2,67 , 1) region code = 0000

Clipp garis EE’ dan garis FF’ karena keduanya invisible, kemudian gambar garis E’F’, karena region code kedua ujungnya 0000

### Clipping garis GH

Garis GH melewati titik G (0, 3) region code 0001 (kiri window) dan H(3, −1) region code 0100

(bawah window)

*m*

3

4

0

3

3

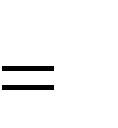
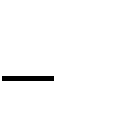
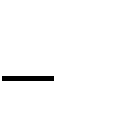
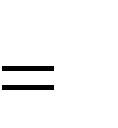
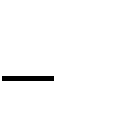
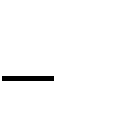
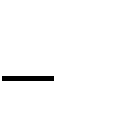
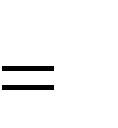
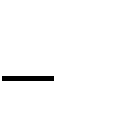
1

1

2

1

2



*x*

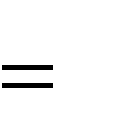
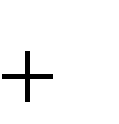
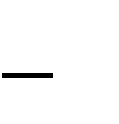
*x*

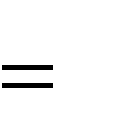
*y*

*y*

Titik potong G’ antara garis GH dengan batas kiri window *x*min = 2 adalah

*y y*1 *m*(*xbatas x*1)



*y* *y*0,3

)

0

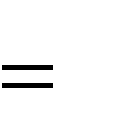
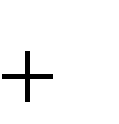
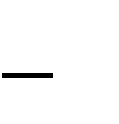
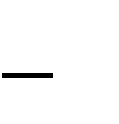
2

(

3

4

3



Titik potong G’(2, 0,3) region code = 0100 ≠ 0000

Titik potong H’ antara garis GH dengan batas bawah window *y*min = 1 adalah

*x*

5

,

1

3

/

4

3

1

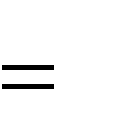
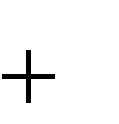
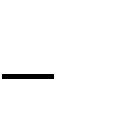
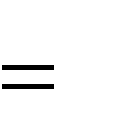
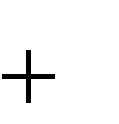
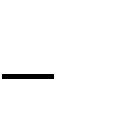
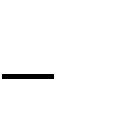
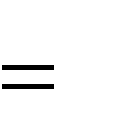
0

)

(

1

1



*x*

*m*

*y*

*y*

*x*

*x*

*batas*

Titik potong H’ (1,5 , 1) region code = 0001 ≠ 0000 karena region code kedua titik potongnya ≠ 0000, maka garis G’H’ invisible. Hasil Clipping

**C’**

**E**

**’**

**F**

**’**

•

•

•

**Window**

1

10

A(3,4)

B(5,9)

D(7,8)

2 8