

# **TUGAS UTS GRAFIKA**

## **IF3260 – GRAFIKA KOMPUTER**

Oleh

Adi Purnama	13514006
Taufic Leonardo Sutejo	13514022
Harry Alvin Waidan Kefas	13514036
Nugroho Satriyanto	13514038
Naufal Malik Rabbani	13514052
Martino Christanto Khuangga	13514084
Luthfi Kurniawan	13514102



PROGRAM STUDI TEKNIK INFORMATIKA  
SEKOLAH TEKNIK ELEKTRO DAN INFORMATIKA  
INSTITUT TEKNOLOGI BANDUNG  
BANDUNG  
2017

# Pendahuluan

Pada UTS matakuliah IF3260 Grafika Komputer, kami diminta untuk membuat sebuah aplikasi yang dibuat pada framebuffer. Kriteria tugas yang harus dipenuhi adalah

1. Implementasi menggambar pada framebuffer
2. Implementasi membuat line
3. Implementasi membuat polygon
4. Implementasi pewarnaan polygon
5. Implementasi rotasi dan translasi
6. Implementasi Clipping dan scaling
7. Implementasi Overlapping

## Dokumentasi Fungsi

### [Line](#)

Pada program ini berfungsi sebagai *support* yang dapat digunakan untuk membuat sebuah garis dan polygon.

### [Rotate](#)

Pada program ini berfungsi untuk memutar sebuah point atau beberapa point dengan acuan putar.

### [ITB\\_Maps\\_View](#)

Pada program ini terdapat 3 hal utama yang dilakukan oleh program. Pertama program menampilkan splash screen yang bertulisan ITB. Kedua program menggambar peta ITB mulai dari gedung, pohon serta jalanan pada ITB dalam bentuk 2D. Ketiga program menimplementasi clipping window pada peta ITB dan scaling (zoom in dan zoom out).

### [Clipping](#)

Pada program ini berfungsi untuk memotong bagian yang berada didalam sebuah bidang.

### [Floodfill](#)

Pada program ini berfungsi untuk menembak firepoint untuk mengisi warna pada sebuah polygon.

### [raster\\_fill](#)

Program ini berfungsi untuk mewarnai gedung dengan teknik algoritma raster

### [Keyevent](#)

Pada program ini berfungsi sebagai key event. Jika suatu tombol ditekan maka akan melakukan sesuatu.

Program ini terdiri dari beberapa modul & ADT. Diantaranya

#### 1. ADT Point

ADT untuk mengelola struktur data titik.

No	Nama	Jenis	Komponen / Keterangan
1.	Point	Tipe Bentuk	int x (absis) int y (ordinat)
2.	Point makePoint(int x, int y)	Fungsi	Membuat point baru

#### 2. ADT Queue of Point

ADT untuk mengelola struktur data queue yang berelemen Point.

No	Nama	Jenis	Komponen / Keterangan
1.	qi_p	Tipe Bentuk	Pointer ke queue Item
2.	queueItem	Tipe Bentuk	Elemen queue qi_p next (next element) Point data (elemen queue)
3.	queue	Tipe Bentuk	queueItem* first (first item) queueItem* last (last item)
4.	void initQueue(queue* q);	Prosedur	Inisialisasi queue
5.	char queueEmpty(queue* q);	Fungsi	Cek apakah queue kosong
6.	void insertPoint(queue* q, Point d);	Prosedur	Menambahkan sebuah point ke dalam queue.
7.	Point nextPoint(queue* q);	Fungsi	Pop elemen queue. Menghapus elemen terdepan saat ini, mengembalikan elemen dibelakangnya.

### 3. ADT Color

ADT untuk mengelola struktur data warna RGB (*red, green, blue*)

No	Nama	Jenis	Komponen / Keterangan
1.	Color	Tipe Bentuk	int R (Intensitas merah , 0-255) int G (Intensitas hijau, 0-255) int B (Intensitas biru, 0-255)
2.	Color setColor(int r, int g, int b);	Fungsi	Membuat warna baru.
3.	int isColorSame(Color C1, Color C2);	Fungsi	Cek apakah kedua warna sama.

### 4. Modul Framebuffer

Modul untuk mengelola hubungan program dengan framebuffer pada linux.

Dependensi : ADT Color & ADT Point

No	Nama	Jenis	Komponen / Keterangan
1.	int fbfd;	Variabel	<i>Framebuffer filedescriptor.</i> Digunakan untuk membuka file <i>framebuffer</i> linux pada <i>"/dev/fb0"</i>
2.	struct fb_var_screeninfo vinfo;	Variabel	Menyimpan <i>variable screeninfo</i> dari <i>framebuffer</i> .
3.	struct fb_fix_screeninfo finfo;	Variabel	Menyimpan <i>fixed screeninfo</i> dari <i>framebuffer</i> .
4.	long int screensize;	Variabel	Ukuran layar (panjang * lebar) , dalam pixel.
5.	char *fbp;	Variabel	Framebuffer dalam memori
6.	void initScreen();	Prosedur	Inisialisasi koneksi dengan framebuffer
7.	void printBackground(Color C);	Prosedur	Mewarnai keseluruhan layar menggunakan warna C

8.	void setXY (int squareSize, int x, int y, Color C);	Prosedur	Mewarnai 1 piksel layar pada koordinat x & y dengan warna C
9.	Color getXY(int x, int y);	Fungsi	Mengembalikan warna piksel pada koordinat layar x & y.
10.	void terminate();	Prosedur	Terminasi koneksi dengan framebuffer.

## 5. Modul Geometri

Modul untuk menggambar objek geometri dasar pada layar, seperti garis, lingkaran, polyline, dan polygon. Algoritma garis dan lingkaran diimplementasikan menggunakan algoritma bresenham.

Dependensi : ADT Color & ADT Point

No	Nama	Jenis	Komponen / Keterangan
1.	void drawBresenhamLine (Point P1, Point P2, Color C, int W);	Prosedur	Menggambar garis dari titik P1 ke P2, dengan warna C, dan ketebalan W. Implementasi menggunakan algoritma bresenham.
2.	void drawPolyline (int n, Point *P, Color C, int W);	Prosedur	Menggambar polyline n titik , dengan warna C dan ketebalan W.
3.	void drawPolygon (int n, Point *P, Color C, int W);	Prosedur	Menggambar poligon n titik, dengan warna C, dan ketebalan W.
4.	void drawCircle (int radius, Point P, int W, Color C);	Prosedur	Menggambar lingkaran dengan jari jari <i>radius</i> , di titik P, dengan warna C dan ketebalan W.

## 6. Modul Transformasi

Modul untuk melakukan transformasi pada objek geometri.

Dependensi : ADT Point

No	Nama	Jenis	Komponen / Keterangan
1.	Point rotatePoint(Point p ,Point pivot, float angle);	Fungsi	Rotasi titik <b>p</b> terhadap titik <i>pivot</i> sebesar <i>angle</i> derajat.

2.	void ScaleLine(Point * p , double scalingFactorX , double scalingFactorY );	Prosedur	Transformasi penskalaan dengan faktor penskalaan sumbu x dan sumbu ya
3.	void TranslationLine(Point * p , int xTranslation, int yTranslation);	Prosedur	Transformasi translasi.

## 7. Modul Color Fill

Modul untuk melakukan pewarnaan objek geometri.

Dependensi : ADT Point , ADT Color, ADT Queue of Point, Modul framebuffer

No	Nama	Jenis	Komponen / Keterangan
1.	void floodFill(int fp_x, int fp_y, Color C, Color fc);	Prosedur	Melakukan pewarnaan menggunakan algoritma flood fill pada <i>firepoint</i> fp_x, fp_y dengan warna C
2.	void raster_fill(int y_min, int y_max, int x_min, int x_max);	Prosedur	Melakukan pewarnaan menggunakan algoritma raster (scanline fill algorithm) di area yang dibatasi oleh y_min, y_max, x_min, dan x_max dengan warna sesuai dengan warna bangunan.

## 8. Modul Clipping

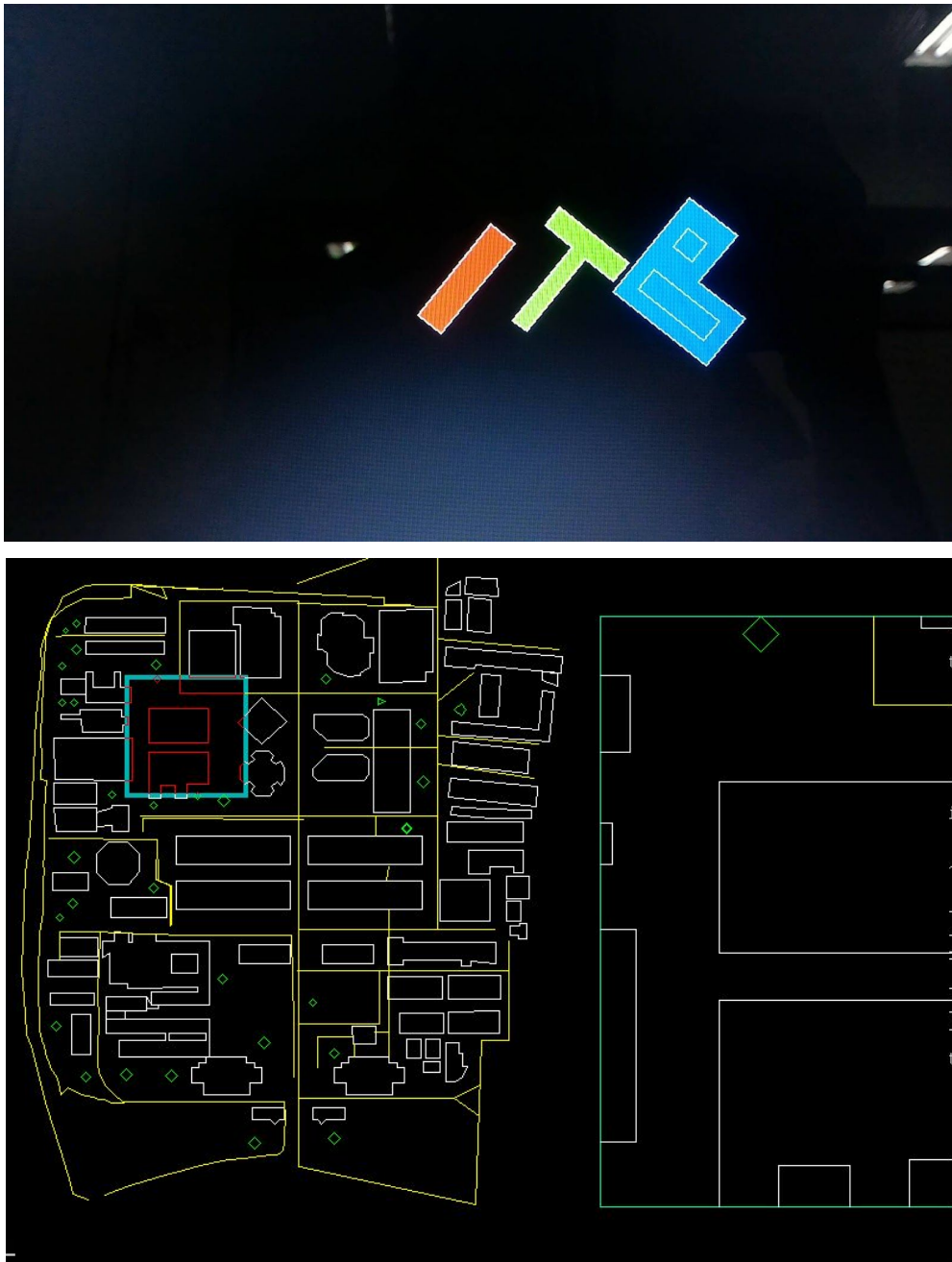
Modul untuk melakukan clipping , menggunakan algoritma Cohen-Sutherland

No	Nama	Jenis	Komponen / Keterangan
1.	ClippingWindow;	Tipe Bentuk	Definisi Clipping Window int xLeft; int xRight; int yTop; int yBottom;
2.	RegionCode;	Tipe Bentuk	Tipe bentuk untuk merepresentasikan region code.
3.	LineAnalysisResult	Tipe Bentuk	Hasil analisis garis terhadap clipping window

			int resultStatusCode; int isLeftIntersection; int isRightIntersection; int isTopIntersection; int isBottomIntersection; int isStartPointCompletelyInside; int isEndPointCompletelyInside; Point startPoint; Point endPoint; RegionCode startPointRegionCode; RegionCode endPointRegionCode;
4.	ClippingWindow setClippingWindow(int left, int right, int top, int bottom);	Fungsi	Inisialisasi clipping window
5.	RegionCode initRegionCode();	Fungsi	Inisialisasi region code.
6.	RegionCode computeRegionCode(Point p, ClippingWindow cw);	Fungsi	Menghitung region code sebuah titik
7.	LineAnalysisResult analyzeLine(Point startPoint, Point endPoint, ClippingWindow cw);	Fungsi	Menganalisis garis terhadap clipping window
8.	LineAnalysisResult initLineAnalysisResult();	Fungsi	Inisialisasi Line Analysis Result
9.	void clipLine(LineAnalysisResult lar1, ClippingWindow cw1 , Point * output);	Prosedur	Melakukan clipping line terhadap clipping window

## Deskripsi Program

Program dibuat dengan bahasa C. Pada awal program ketika dijalankan program akan menampilkan sebuah tampilan splash screen yang bertulisan ITB. Tulisan tersebut akan bergerak dari kanan ke kiri dan akan mantul kembali dari kiri ke kanan. Hal tersebut berlangsung selama 3 kali dan program akan lanjut menampilkan peta ITB. Berikut adalah contoh tampilan peta ITB.



Pada tampilan peta ITB terdapat 2 window. Window yang lebih kecil dengan stroke yang berwarna biru berfungsi sebagai clipping window sedangkan window yang lebih besar berfungsi sebagai hasil clipping.



untuk menjalankan program tersebut dapat dilakukan dengan cara:

1. ctrl + alt + F2 pada ubuntu. Pastikan terdapat /dev/fb0 pada ubuntu anda.
2. masukan username dan password anda.
3. Lakukan sudo su
4. Untuk kompilasi, lakukan make
5. Jalankan program yang bernama itbmapview dengan cara ./itbmapview
6. Terdapat window kecil yang berfungsi sebagai *zoom map*. Untuk menggeser window tersebut dapat dilakukan dengan cara menekan tombol *up, down, left, right*. Untuk memperbesar window tersebut dapat dilakukan dengan cara menekan tombol +, -.
7. Untuk keluar dari program dapat dilakukan dengan cara alt + F7

## Algoritma yang Digunakan

- Lining menggunakan algoritma Bresenham.
- clipping menggunakan algoritma Cohen-Sutherland
- overlapping menggunakan algoritma Painter.
- coloring menggunakan algoritma floodfill.

## Pembagian Tugas

Nama	Kerja
Adi Purnama	Refractoring, Dokumentasi modul
Taufic Leonardo Sutejo	Rotate Splash Screen
Harry Alvin Waidan Kefas	Overlapping
Nugroho Satriyanto	Rotate windows
Naufal Malik Rabbani	Overlapping
Martino Christanto Khuangga	ITB Splash Screen
Luthfi Kurniawan	ITB Splash Screen