

## BAB 2

# PRIMITIVE OBJECTS

### A. KOMPETENSI DASAR

- Memahami jenis-jenis primitive objects.
- Memahami dan dapat membuat objek primitif.

### B. ALOKASI WAKTU

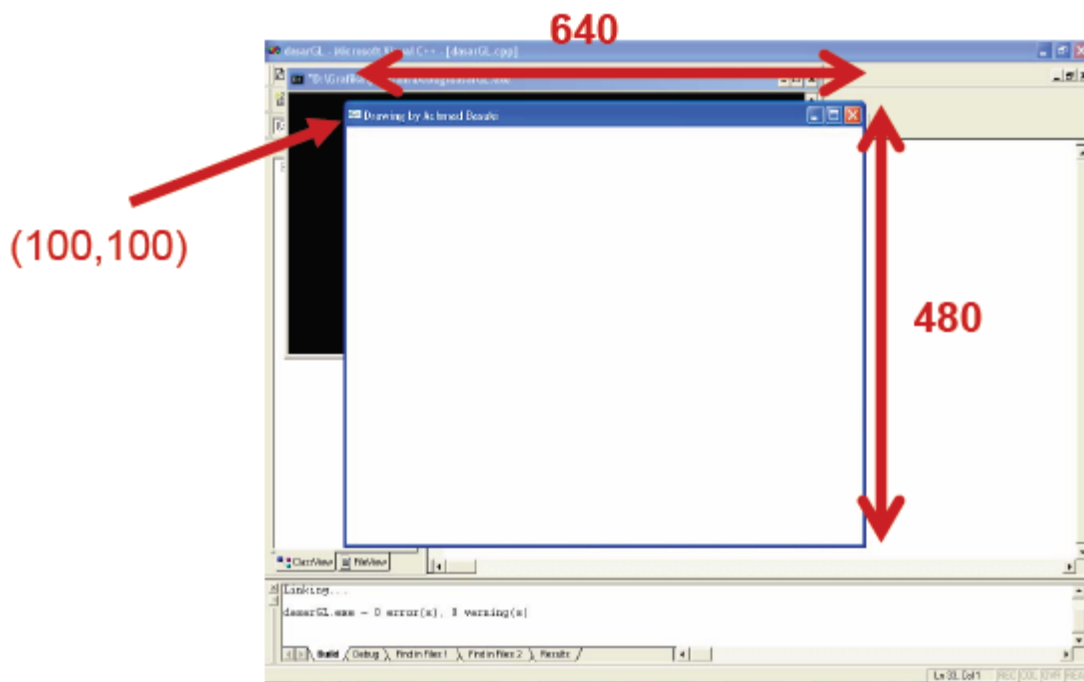
2 js (2x50 menit)

### C. PETUNJUK

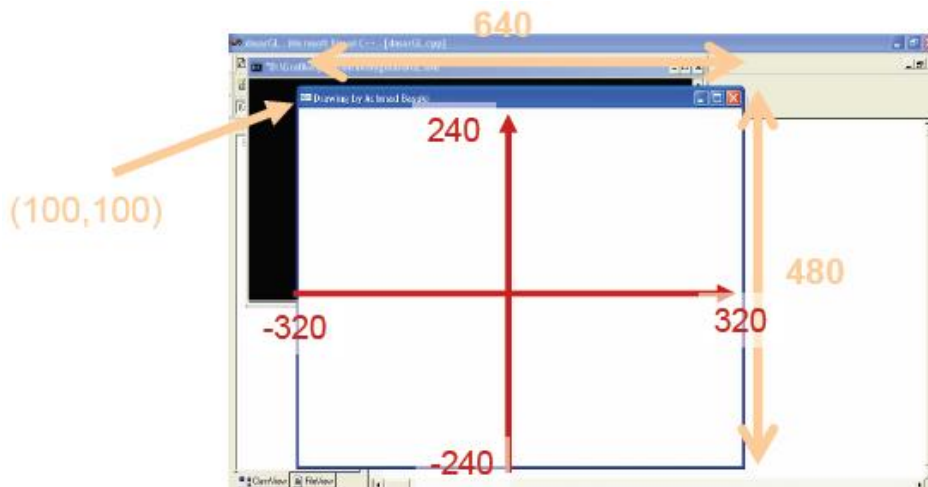
- Awali setiap aktivitas dengan do'a, semoga berkah dan mendapat kemudahan.
- Pahami Tujuan, dasar teori, dan latihan-latihan praktikum dengan baik dan benar.
- Kerjakan tugas-tugas dengan baik, sabar, dan jujur.
- Tanyakan kepada asisten/dosen apabila ada hal-hal yang kurang jelas.

### D. DASAR TEORI

#### 1. Sistem Koordinat OpenGL



2. Fungsi `gluOrtho2D(left,right,bottom,top) → gluOrtho2D(-320,320,-240,240)`



3. Fungsi dasar menggambar titik `→ glVertex?()`
- `glVertex2i(x,y)` → Menggambar titik pada koordinat x dan y, dengan nilai satuan berupa **integer**. Contoh `glVertex2i(10,10);`
  - `glVertex2f(x,y)` → Menggambar titik pada koordinat x dan y, dengan nilai satuan berupa **float**. Contoh `glVertex2f(10.0,10.0);`
  - `glVertex2d(x,y)` → Menggambar titik pada koordinat x dan y, dengan nilai satuan berupa **double**. Contoh `glVertex2d(10.0,10.0);`
4. Objek primitif
- Titik `→ GL_POINTS`
  - Garis `→ GL_LINES`
  - Poliline `→ GL_LINE_STRIP`
  - Poligon (no fill) `→ GL_LINE_LOOP`

## E. AKTIFITAS KELAS PRAKTIKUM

Berikut adalah script dasar untuk kegiatan praktikum selanjutnya

```
glutInitWindowPosition(100,100);  
glutInitWindowSize(320,240);  
    dan  
gluOrtho2D(-320.,320.,-240.,240.);
```

1. Buatlah project baru pada Visual Studio dengan nama **prak1-Points**. Dan tambahkan fungsi callback untuk **glutDisplayFunc(drawDot);**. Tampilkan hasilnya berupa screenshot.

```
void drawDot() {  
    glClear(GL_COLOR_BUFFER_BIT);  
    glBegin(GL_POINTS);  
        glVertex2i(x0,y0);  
        glVertex2i(x1,y1);  
        glVertex2i(x2,y2);  
    glEnd();  
    glFlush();  
}
```

2. Buatlah project baru pada Visual Studio dengan nama **prak1-PointSize**. Dan tambahkan fungsi **glPointSize(int size);** untuk mengubah ukuran objek Points. Pindah-pindahkan **glPointSize** pada A, B, dan C. Sajikan Screen shot hasil A, B, C dan simpulkan.

```
//[A]
void drawDot() {
    glClear(GL_COLOR_BUFFER_BIT);
    //[B]
    glBegin(GL_POINTS);
        glVertex2i(x0,y0);
        //[C]
        glVertex2i(x1,y1);
        glVertex2i(x2,y2);
    glEnd();
    glFlush();
}
```

3. Buatlah project baru pada Visual Studio dengan nama **prak1-Lines**. Dan tambahkan fungsi callback untuk **glutDisplayFunc(drawLine);** . Tampilkan hasilnya berupa screenshot.

```
void drawLine() {
    glClear(GL_COLOR_BUFFER_BIT);
    glBegin(GL_LINES);
        glVertex2d(x0,y0);
        glVertex2d(x1,y1);
    glEnd();
    glFlush();
}
```

4. Buatlah project baru pada Visual Studio dengan nama **prak1-LineWidth**. Dan tambahkan fungsi **glLineWidth(int size);** untuk mengubah ukuran objek Points. Pindah-pindahkan **glLineWidth** pada A, B, dan C. Sajikan Screen shot hasil A, B, C dan simpulkan.

```
void drawLine() {
    glClear(GL_COLOR_BUFFER_BIT);
    //[A]
    glBegin(GL_LINES);
        glVertex2d(x0,y0);
        //[B]
        glVertex2d(x1,y1);
        //[C]
    glEnd();
    glFlush();
}
```

5. Buatlah project baru pada Visual Studio dengan nama **prak1-Poliline**. Buatlah agar hasil yang ditampilkan membentuk **persegi**. Sajikan screenshotnya.

```
void drawPoliline() {
    glClear(GL_COLOR_BUFFER_BIT);
    glBegin(GL_LINE_STRIP);
        glVertex2d(x0,y0);
        glVertex2d(x1,y1);
        glVertex2d(x2,y2);
        glVertex2d(x3,y3);
    glEnd();
    glFlush();
}
```

6. Buatlah project baru pada Visual Studio dengan nama **prak1-Poligon**. Buatlah agar hasil yang ditampilkan membentuk **persegi**. Sajikan screenshotnya.

```
void drawPoligon() {  
    glClear(GL_COLOR_BUFFER_BIT);  
    glBegin(GL_LINE_LOOP);  
        glVertex2d(x0,y0);  
        glVertex2d(x1,y1);  
        glVertex2d(x2,y2);  
    glEnd();  
    glFlush();  
}
```