

# MODUL 7

## INTERAKSI DENGAN MOUSE

### A. KOMPETENSI DASAR

- Memahami prinsip-prinsip pembuatan interaksi objek 2D menggunakan mouse.
- Membuat interaksi objek 2D menggunakan mouse.

### 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. Fungsi dasar pembuatan interaksi dengan menggunakan *MouseFunction*

```
GLUTAPI void APIENTRY glutMouseFunc(void (GLUTCALLBACK *func) (int  
button, int state, int x, int y));
```

Paramater *func* adalah fungsi yang akan ditangani dengan event klik mouse.

```
GLUTAPI void APIENTRY glutMotionFunc(void *func (int x, int y));
```

Fungsi di atas adalah fungsi pelengkap dari fungsi interaksi mouse untuk mendeteksi gerakan mouse.

#### 2. Inisialisasi dalam penggunaan *MouseFunction*

```
void mouse(int button, int state, int x, int y) {  
    if(button==GLUT_LEFT_BUTTON && state==GLUT_DOWN)  
        drawDot(x, 480-y);  
    if(button==GLUT_RIGHT_BUTTON && state==GLUT_DOWN)  
        drawDot2(x, 480-y);  
    if(button==GLUT_MIDDLE_BUTTON && state==GLUT_DOWN)  
        drawDot3(x, 480-y);  
}  
  
void motion(int x, int y) {  
  
}
```

GLUT\_LEFT\_BUTTON untuk inialisasi button mouse kiri.  
GLUT\_RIGHT\_BUTTON untuk inialisasi button mouse kanan.  
GLUT\_MIDDLE\_BUTTON untuk inialisasi button mouse tengah.  
Fungsi dari GLUT\_DOWN adalah untuk inialisasi ketika tombol mouse ditekan.  
Fungsi dari GLUT\_UP adalah untuk inialisasi ketika tombol mouse dilepaskan.

Kemudian pada *main* program perlu menambahkan fungsi untuk *callback* fungsi *MouseFunction*.

```
glutMouseFunc(mouse);  
glutMotionFunc(motion);
```

## E. AKTIVITAS KELAS PRAKTIKUM

Berikut adalah script dasar untuk kegiatan praktikum selanjutnya.

```
int w = 640, h = 640; //variabel global  
glutInitWindowSize(w, h);  
gluOrtho2D(-w/2, w/2, -h/2, h/2);
```

1. Buatlah project baru pada Visual Studio dengan nama **prak7-mouseMotion**. Fungsi **myDisplay** *callback function* yang di dalamnya menggunakan fungsi Points untuk memindah posisi objek sesuai *pointer mouse*.

```
float x=0, y=0, z=0;  
void myDisplay(void)  
{  
    glClear(GL_COLOR_BUFFER_BIT);  
    glPushMatrix();  
        glTranslatef(x, y, z);  
        drawQuad(5, 5);  
    glPopMatrix();  
    glFlush();  
}
```

Fungsi mouse untuk inialisasi *mouse event*.

```
void mouse(int button, int state, int xmouse, int ymouse)  
{  
    if(button==GLUT_LEFT_BUTTON && state==GLUT_DOWN){  
        x = xmouse-(w/2);  
        y = (h/2)-ymouse;  
    }  
}
```

2. Sempurnakan program di atas sehingga tombol **tengah** ditekan akan menyebabkan objek memiliki skala yang **membesar** (1.1) dan tombol **kanan** ditekan akan menyebabkan skala **mengecil** (0.9). Berikan **source code** program.
3. Ubahlah program sehingga tombol **kanan** ditekan skala **membesar** (2.0) dan ketika tombol **kanan dilepaskan (UP)** skala mengecil (0.5). Berikan *source code* program.

4. Buatlah nilai parameter yang ada pada fungsi  
`gluOrtho2D(-100, 100, -100, 100);`  
 Jelaskan apa yang terjadi dan beri kesimpulan (**gambar sistem koordinat**) hubungan antara sistem koordinat OpenGL dan sistem koordinat yang dianut oleh *mouse pointer*.
5. Buatlah project baru pada Visual Studio dengan nama **prak7-mouseMotion1**. Gunakan fungsi yang sama dengan **No.1** dan lengkapi dengan fungsi **Motion**. Buatlah fungsi untuk deteksi motion. **Berikan kesimpulan!**  

```
void motion(int xmouse, int ymouse)
{
    x = xmouse - (w/2);
    y = (h/2) - ymouse;
}
```
6. Sempurnakan program di atas sehingga ketika **tombol kiri ditekan** dan **mouse digerakkan** maka akan terjadi proses **drag** dengan memanfaatkan event **DOWN** dan flag untuk proses drag ini. Berikan **source code** program. **Beri kesimpulan!**
7. Sempurnakan lagi program di atas sehingga ketika tombol kiri ditekan maka objek akan **membesar** (2.0) dan proses drag dimulai, ketika tombol kiri dilepaskan kembali maka objek akan **drop** dan skala objek menjadi normal kembali atau **mengecil** (0.5) (dengan memanfaatkan event **DOWN/UP** dan flag untuk drag and drop).  
 Tampilkan source codenya. **Berikan kesimpulan!**

## F. TUGAS

1. Buatlah program yang dapat mengubah **skala** objek secara interaktif menggunakan interaksi **drag**. Ketika **tombol kanan ditekan** maka posisi **x** dan **y** disimpan dalam **variabel global**, jika drag dilakukan maka **jarak pointer** terhadap **posisi** yang disimpan sebelumnya akan menyebabkan **perubahan skala** secara interaktif (**menjauh = membesar, mendekat = mengecil**). Selain itu objek juga akan beranimasi **berputar** pada saat kondisi **drag** dan **berhenti** berputar saat **dilepas**.