

# KONPUTAGAILU BIDEZKO GRAFIKOAK



*Proiektua: 2. Fasea*



Universidad  
del País Vasco

Euskal Herriko  
Unibertsitatea

INFORMATIKA  
FAKULTATEA  
FACULTAD  
DE INFORMÁTICA

Objektuen  
aldaketak (bira-  
keta, traslazioa,  
tamainaren alda-  
keta) inplemen-  
tatu

## 2. FASEKO EGINKIZUNAK



Objektuen  
aldaketak (bira-  
keta, traslazioa,  
tamainaren alda-  
keta) inplemen-  
tatu

Aldaketak  
munduko zein  
objektuko erre-  
ferentzi sisteman  
implementatu

## 2. FASEKO EGINKIZUNAK

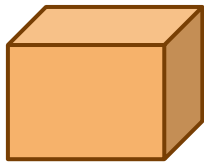


Objektuen  
aldaketak (bira-  
keta, traslazioa,  
tamainaren alda-  
keta) inplemen-  
tatu

Aldaketak  
munduko zein  
objektuko erre-  
ferentzi sisteman  
implementatu

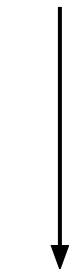
Objektu bakoitza-  
ri eragindako al-  
daketak desegite-  
ko aukera inple-  
mentatu

## 2. FASEKO EGINKIZUNAK



Objektua  
(erpinen koordenatuak)

GL\_MODELVIEW  
GL\_PROJECTION  
glViewport



$P_O$

Aldaketa  
Matrizea



$P_W$

Kamera  
Matrizea



$P_K$

Proiektzio  
Matrizea



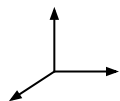
$P_{2D}$

Viewport  
Matrizea

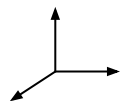


$P_L$

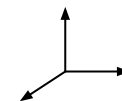
Rasterizazioa



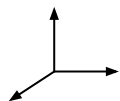
Objektuaren  
ES



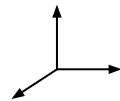
Munduaren  
ES



Kameraren  
ES



ES  
Kanonikoa



Pantailaren  
ES

```
glMatrixMode (GL_MODELVIEW);  
glLoadIdentity ();  
draw_axes ();
```

OpenGL-k barrutik matrizeak erabiltzen ditu. Horietako bat, **ModelView** deitzen da. Matrize hau **objektuen koordinatuak sartzean biderkatzen da**.

```
while (aux_obj != 0) {
```

Objektuak banan-banan marrazten ditugu. Horretarako ...

```
...
```

```
glLoadIdentity();
```

... lehenik identitate matrizea sartzen dugu.

```
for (f=0; f<aux_obj->num_faces; f++) {
```

Gero, aurpegi bakoitzeko ...

```
glBegin(GL_POLYGON);
```

... poligono bat sartzen dugu, bere koordinatuak sartuz.

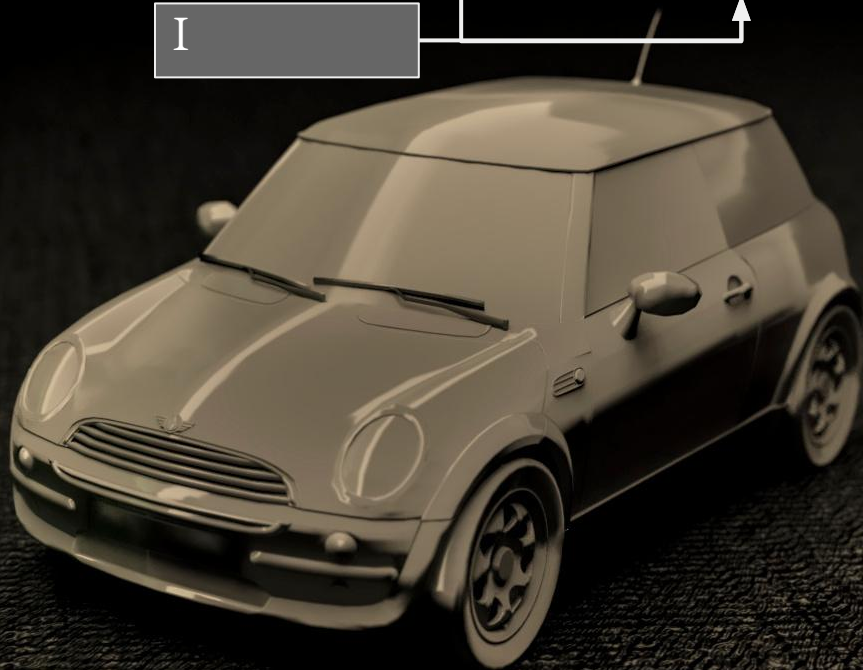
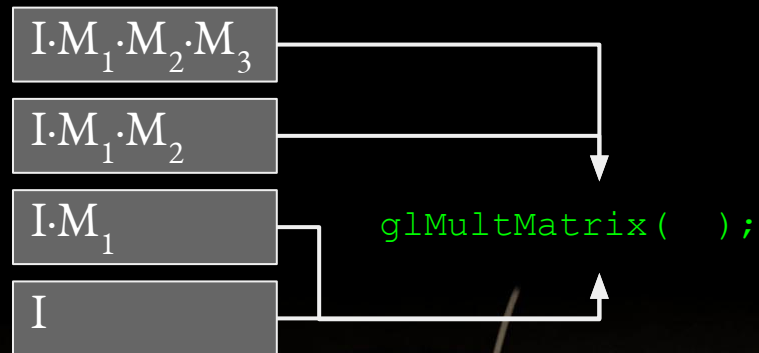
```
for(v=0; v<aux_obj->face_table[f].num_vertices; v++){  
    v_index=aux_obj->face_table[f].vertex_table[v];  
    glVertex3d(aux_obj->vertex_table[v_index].coord.x,  
               aux_obj->vertex_table[v_index].coord.y,  
               aux_obj->vertex_table[v_index].coord.z);  
}
```

Kontutan izan aurpegi bakoitzak bere erpinen **indizea** dituela, ez koordinatuak. Erpin bakoitzeko indizea lortu eta vertex\_table egituratik atera. **Sartzen ditugun koordinatuak ModelView matrizearekin biderkatzen dira**

```
glEnd();
```

```
aux_obj = aux_obj->next;
```

```
}
```



# EGOERA MAKINA



**Biraketa** Aktibatuta



Y ardatzean **biraketa** negatiboa



Z ardatzean **biraketa** positiboa



**Taslazioa** Aktibatuta



X ardatzean **traslazio** negatiboa



Z ardatzean **traslazioa** positiboa



**Tamaina aldaketa** Aktibatuta

...



Biraketa **Aktibatu**



Traslazioa **Aktibatu**



Tamaina aldaketa **Aktibatu**



Aldaketak ardatz batean



Aldaketak beste ardatz batean



Aldaketak beste ardatzean



```
struct tmatrix {
    GLdouble      * current_matrix;
    struct tmatrix * previous_matrix;
};
```

```
struct object3d {
    GLint      num_vertices;
    ...
    tmatrix    * transformation;
    ...
};
```

**F**

## io.c

1. Erreserbatu memoria tmatrix egitura baterako (malloc)
2. auxiliar\_object->transformation = Sortutako egitura
3. Sortu identitate matrize bat esleitu auxiliar\_object->transformation->current\_matrix eremuari
4. auxiliar\_object->transformation->previous\_matrix = 0

**←**

## io.c

1. Sortu aldaketa eragiteko behar den matrizea
2. Biderkatu matrize hori \_selected\_object->transformation->current\_matrix matrizearekin
3. Erreserbatu memoria tmatrix egitura baterako (malloc)
4. Egitura horren current\_matrix = matrizeen biderketa
5. Egitura horren previous\_matrix = \_selected\_object->transformation
6. \_selected\_object->transformation = tmatrix egitura berria

## display.c

1. Objektu marraztean, \_glMultmatrixd(aux\_obj->transformation->current\_matrix)

# KONPUTAGAILU BIDEZKO GRAFIKOAK



## *Proiektua: 2. Fasea*



Universidad  
del País Vasco

Euskal Herriko  
Unibertsitatea

INFORMATIKA  
FAKULTATEA  
FACULTAD  
DE INFORMÁTICA