

P. J. Martín, A. Gavilanes

Departamento de Sistemas Informáticos y Computación

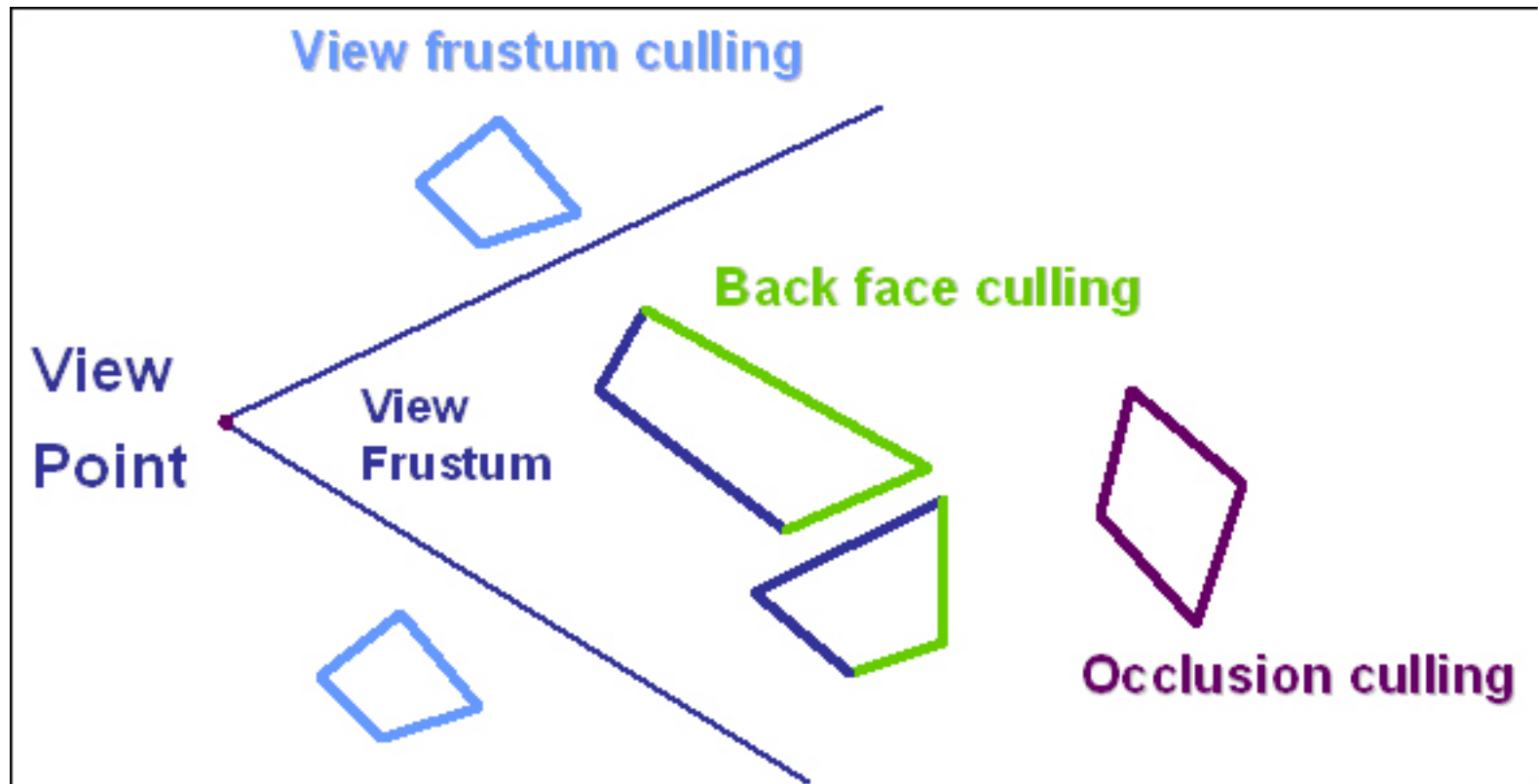
Facultad de Informática

Universidad Complutense de Madrid

- ❑ *Culling*: eliminar elementos de un conjunto enorme; sacrificar selectivamente; entresacar o descartar
- ❑ Tipos principales de *culling*:
 - ❑ *Backface culling*: eliminación de caras traseras
 - ❑ *(view) Frustum culling*: eliminación de caras de una malla que están fuera del *frustum*
 - ❑ *Occlusion culling*: eliminación de objetos que están ocultos por otros objetos
- ❑ Orden de complejidad de los tipos de *culling*:
 - ❑ El *backface culling* opera a nivel de una sola cara
 - ❑ El *frustum culling* opera a nivel de conjuntos de caras
 - ❑ El *occlusion culling* opera a nivel de objetos.
 - ❑ En consecuencia:

BC < FC < OC

donde < representa "es más sencillo que"

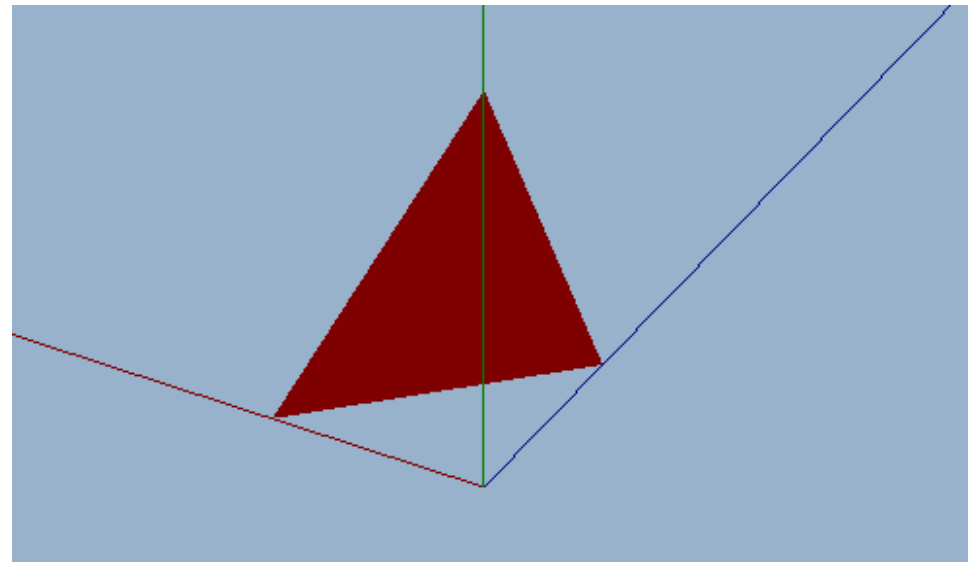
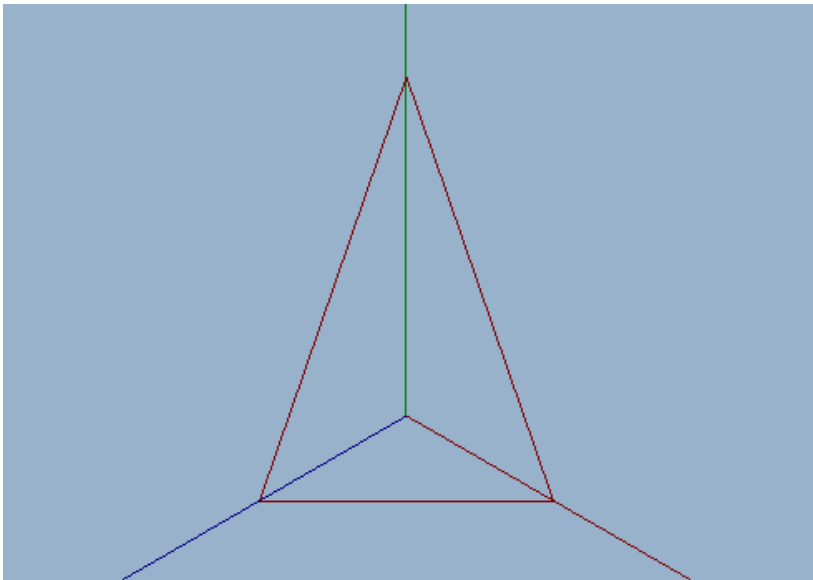


Caras frontales y traseras

- ❑ Los dos lados de una cara son independientes: pueden pintarse de modo y color diferente.
- ❑ Para pintar de modo diferente se usa el comando:

`glPolygonMode(face, mode);`

donde `face` es `GL_FRONT`, `GL_BACK` o `GL_FRONT_AND_BACK`, y `mode` es `GL_FILL`, `GL_LINE` o `GL_POINT`. En este ejemplo, las caras frontales están dibujadas en modo línea (`GL_LINE`) y las caras traseras en modo relleno (`GL_FILL`).



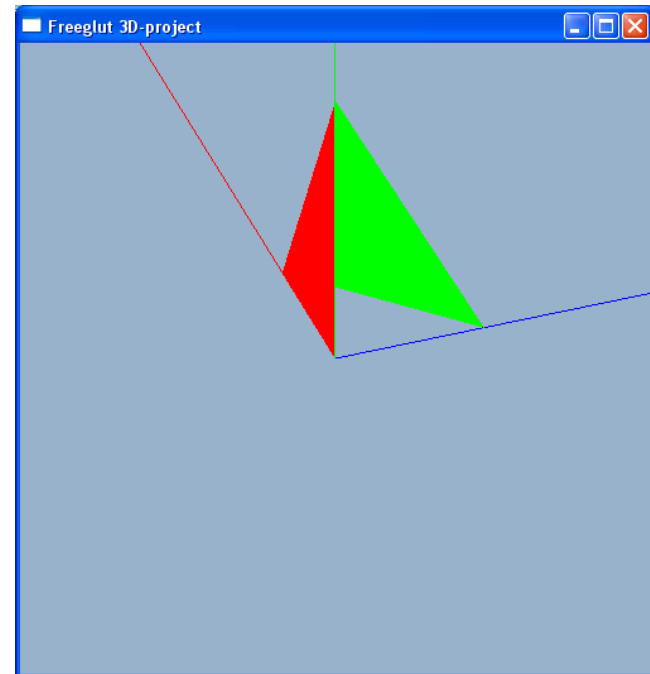
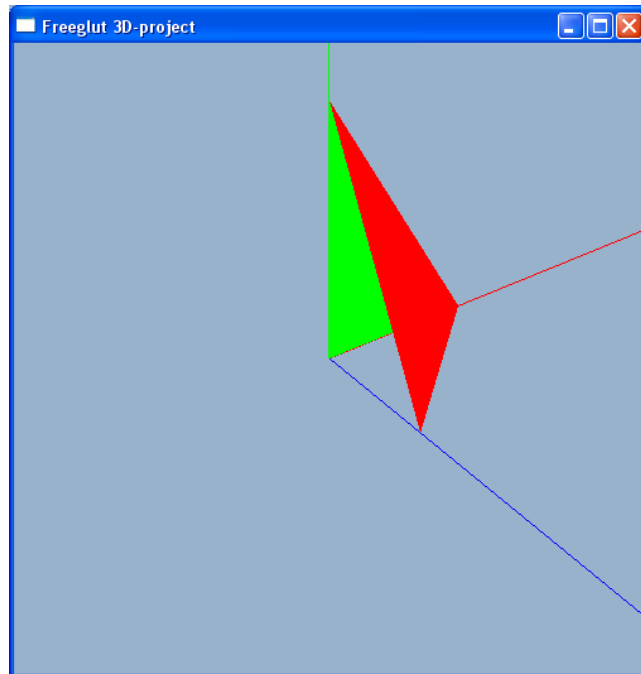
Caras frontales y traseras

- ❑ Para pintar de color diferente se usa el comando:

```
glLightModeli(GL_LIGHT_MODEL_TWO_SIDE, GL_TRUE);
```

que le dice a OpenGL que pinte las caras traseras con los parámetros que se den y que invierta normales para los vértices que aparezcan. Por defecto, el valor del segundo parámetro es GL_FALSE.

- ❑ En el ejemplo, las caras frontales son rojas y las traseras verdes:



- ❑ La especificación del color de los lados de las caras se hace como se hace en OpenGL, es decir, mediante propiedades de los materiales:

```
GLfloat adf[]={1, 0, 0, 1};  
glMaterialfv(GL_FRONT, GL_AMBIENT_AND_DIFFUSE, adf);  
GLfloat adb[]={0, 1, 0, 1};  
glMaterialfv(GL_BACK, GL_AMBIENT_AND_DIFFUSE, adb);
```

o mediante registro de color:

```
glEnable(GL_COLOR_MATERIAL);  
glColorMaterial(GL_FRONT, GL_AMBIENT_AND_DIFFUSE);  
glColor3f(1, 0, 0);  
glColorMaterial(GL_BACK, GL_AMBIENT_AND_DIFFUSE);  
glColor3f(0, 1, 0);
```

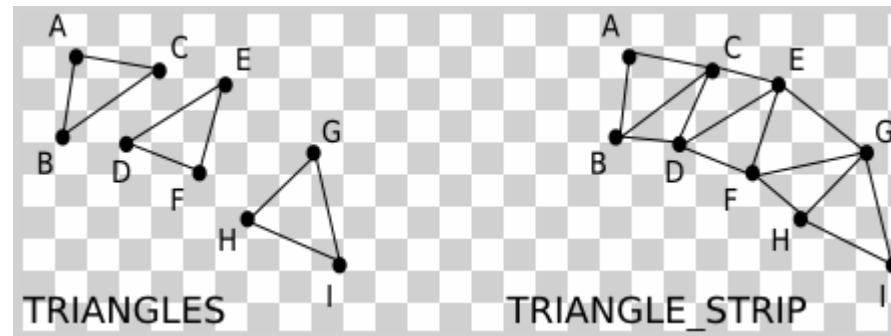
- ❑ Las caras frontales y las caras traseras.
 - ❑ Las caras frontales se proporcionan en sentido contrario a las agujas del reloj (**C**ounter **C**lock**W**ise)
 - ❑ Las caras traseras se dan en sentido horario (**C**lock**W**ise)
 - ❑ El comando

`glFrontFace(constante);`

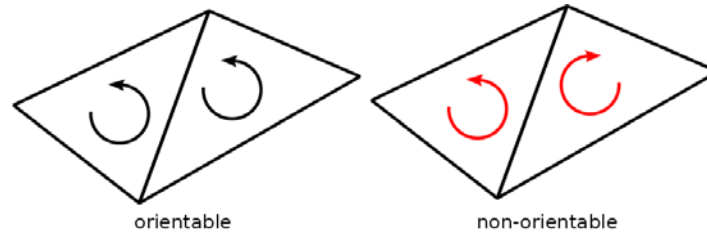
especifica el orden en que se proporcionan los vértices de las caras frontales.

- ❑ Las constantes `GL_CW`, `GL_CCW`.
- ❑ El valor por defecto es `GL_CCW` y por ello los vértices de las caras frontales se deben suministrar, a menos que se especifique lo contrario, en sentido anti-horario.

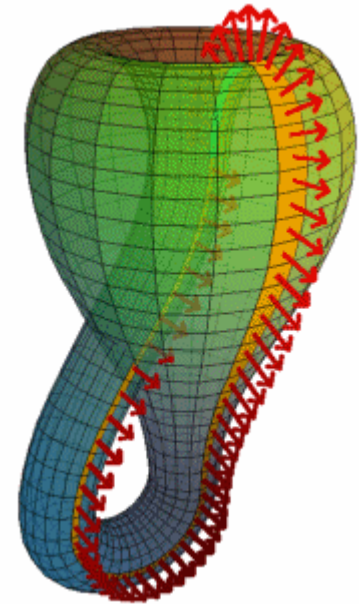
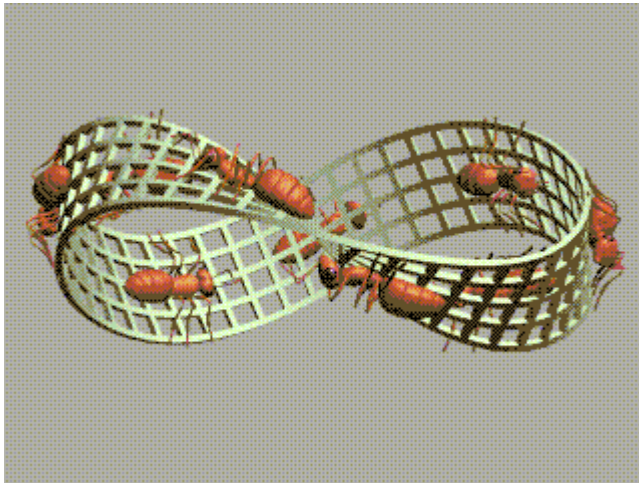
- ❑ El proceso para pintar una cara es el siguiente:
 - ❑ Se determina el orden en que se proporcionan los vértices, que depende del tipo de primitiva de OpenGL que se use:
 - ❑ GL_TRIANGLES:
 - ❑ $A,B,C,D,E,F,G,H,I \Rightarrow \text{triangulo}(A,B,C), \text{triangulo}(D,E,F), \dots$
 - ❑ GL_TRIANGLE_STRIP:
 - ❑ $A,B,C,D,E,F,G,H,I \Rightarrow \text{triangulo}(A,B,C), \text{triangulo}(B,D,C), \dots$



- Triangulaciones consistente e inconsistentemente orientadas.

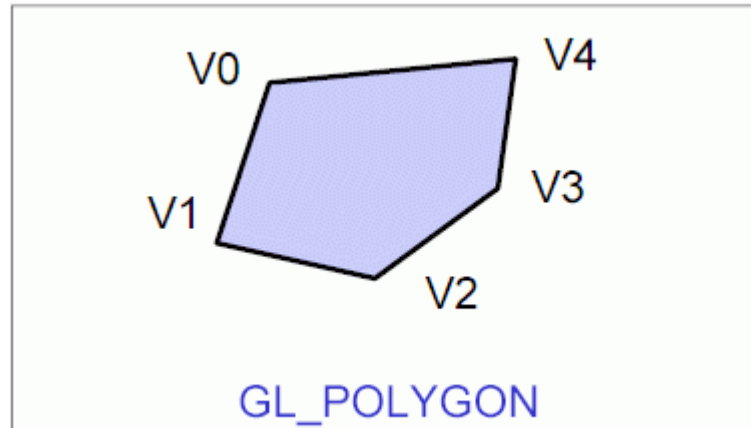


- La banda de Möbius y la botella de Klein no son orientables.



- GL_POLYGON:

- $v_0, v_1, v_2, v_3, v_4 \Rightarrow \text{polígono}(v_0, v_1, v_2, v_3, v_4)$



- Por cada grupo de vértices, se determina si la cámara ve el grupo en sentido horario o anti-horario, identificando, por tanto, si el triángulo o polígono es frontal o trasero.
- Se dibuja el triángulo o polígono, en consecuencia.

- ❑ El back-face culling (o polygon culling) permite no pintar caras frontales, traseras o ambas.
- ❑ Cuando una cara no se dibuja, se ve lo que hay detrás de ella.
- ❑ Los comandos de OpenGL para hacerlo son:

```
glCullFace(face);
```

donde `face` puede ser una de las constantes `GL_FRONT`, `GL_BACK` o `GL_FRONT_AND_BACK`.

- ❑ El culling se activa/desactiva con los comandos:

```
glEnable(GL_CULL_FACE);
```

```
glDisable(GL_CULL_FACE);
```

- ❑ El culling permanece activado hasta que se desactiva.
- ❑ Las escalaciones (negativas) afectan al culling.

❑ Ejemplo:

```
//Dibujo de la tapa rotada
glPushMatrix();
    glTranslatef(0.0, 1.0, -1.0);
    glRotatef((float)step, -1.0, 0.0, 0.0);
    glTranslatef(0.0, -1.0, 1.0);
    glDrawElements(GL_QUADS,4,GL_UNSIGNED_BYTE, &quadIndices[20]);
glPopMatrix();
```

...

```
//Dibujo de la esfera
glutSolidSphere(1.0, 40, 40);
```

- ❑ Este código muestra una caja que contiene una esfera. Al abrir la tapa se ve su cara trasera, pero también se están pintando las caras traseras de la esfera, que no se ven.

❑ Ejemplo:

```
//Dibujo de la tapa rotada
glPushMatrix();
    glTranslatef(0.0, 1.0, -1.0);
    glRotatef((float)step, -1.0, 0.0, 0.0);
    glTranslatef(0.0, -1.0, 1.0);
    glDrawElements(GL_QUADS,4,GL_UNSIGNED_BYTE, &quadIndices[20]);
glPopMatrix();
```

...

```
//Culling de las caras traseras de la esfera
glEnable(GL_CULL_FACE);
glCullFace(GL_BACK);
```

```
//Dibujo de la esfera
glutSolidSphere(1.0, 40, 40);
```

- ❑ Este código hace que no se pinten las caras traseras de la esfera, pero al abrir la tapa, no se ven las caras internas de la caja, en particular, la cara trasera de la tapa.

❑ Ejemplo:

```
//Dibujo de la tapa rotada
glPushMatrix();
    glTranslatef(0.0, 1.0, -1.0);
    glRotatef((float)step, -1.0, 0.0, 0.0);
    glTranslatef(0.0, -1.0, 1.0);
    glDrawElements(GL_QUADS,4,GL_UNSIGNED_BYTE, &quadIndices[20]);
glPopMatrix();
...
//Culling de las caras traseras de la esfera
glEnable(GL_CULL_FACE);
glCullFace(GL_BACK);

//Dibujo de la esfera
glutSolidSphere(1.0, 40, 40);

//Se desactiva el culling
glDisable(GL_CULL_FACE);
```

- ❑ Este código permite que no se pinten las caras traseras de la esfera, pero sí las de la caja, en particular, la de la tapa.

❑ Ejemplo:

```
//Culling de las caras frontales de la caja
glEnable(GL_CULL_FACE);
glCullFace(GL_FRONT);

//Dibujo de la caja con la tapa rotada
glPushMatrix();
...
glDrawElements(GL_QUADS,4,GL_UNSIGNED_BYTE, &quadIndices[20]);
glPopMatrix();
...
//Culling de las caras traseras de la esfera
glEnable(GL_CULL_FACE);
glCullFace(GL_BACK);

//Dibujo de la esfera
glutSolidSphere(1.0, 40, 40);

//Se desactiva el culling
glDisable(GL_CULL_FACE);
```

- ❑ Este código permite que se pinten las caras traseras de la caja, pero no las de la esfera (aplicación en videojuegos).