

Lezione 9

Picking

1

Interazione con gli oggetti grafici

- ❖ Affinché l'utente possa interagire con le primitive che formano la scena visualizzata, è necessaria una cooperazione tra OpenGL e il toolkit che gestisce l'interfaccia grafica (GLUT per noi):
 - ❖ Il toolkit gestisce gli eventi generati dall'utente
 - ❖ OpenGL trasferisce l'effetto di questi eventi sulle entità grafiche
- ❖ È necessario che l'utente possa *selezionare* la/le entità di proprio interesse tra quelle visualizzate
- ❖ Questa operazione viene detta *picking*

2

Picking

In breve, deve fornire la funzionalità seguente:

- ❖ L'utente con il mouse fa click sull'area di visualizzazione
- ❖ Al programma viene comunicato quali primitive/ oggetti sono visualizzate in una determinata (piccola) area intorno al puntatore del mouse

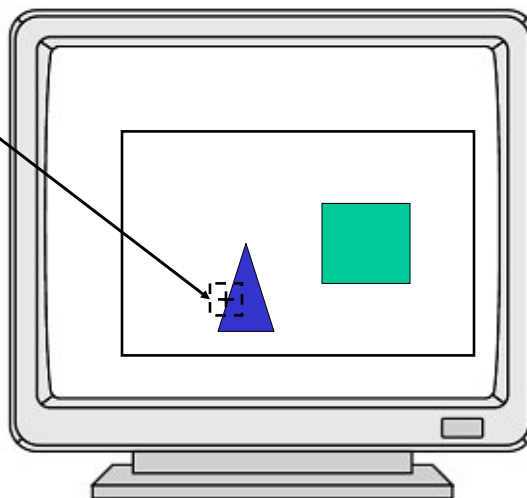
3

Picking

Finestra di pick

Vengono considerate le primitive visualizzate nella finestra di pick :

- ❖ triangolo blu sì
- ❖ rettangolo verde no



4

Picking

Cosa è necessario:

- ❖ Capire dov'è il puntatore (ci pensa GLUT)
- ❖ Capire quali sono le primitive selezionate (ci pensa OpenGL)
- ❖ Comunicarle al programma:
 - ❖ serve un modo per identificare le primitive
 - ❖ si da un "nome" ad ogni primitiva / gruppo di primitive

5

Il meccanismo dei nomi

- ❖ Ogni primitiva visualizzata può ottenere un *nome* (numero intero)
- ❖ OpenGL mantiene un intero *stack* di nomi, al fine di gestire situazioni gerarchiche (simile a quanto fa con le matrici)
- ❖ Il *nome corrente* corrisponde a quello che si trova al top dello stack
- ❖ Tutte le primitive che vengono visualizzate sono etichettate con il nome corrente

6

Il meccanismo dei nomi

Lo stack dei nomi si gestisce con le primitive usuali:

- ❖ `glInitNames()` inizializza lo stack a vuoto
- ❖ `glPushName(<name>)` inserisce un nuovo nome al top dello stack
- ❖ `glPopName()` toglie il nome al top dello stack (facendo emergere il nome subito sotto)
- ❖ `glLoadName(<name>)` rimpiazza il top dello stack con un nome nuovo (attenzione: lo stack non deve essere vuoto)

7

Il meccanismo dei nomi

I nomi vanno assegnati mentre si visualizzano le primitive

❖ Ad esempio:

```
glInitNames();  
glPushName(1);  
// disegna il primo gruppo di primitive  
glLoadName(2);  
// disegna il secondo gruppo di primitive  
glLoadName(3);  
// disegna il terzo gruppo di primitive  
// disegna il quarto gruppo di primitive
```

in questo modo:

- ❖ tutte le primitive del primo gruppo hanno nome 1, tutte quelle del secondo gruppo hanno nome 2, tutte quelle del terzo e tutte quelle del quarto hanno nome 3

8

Il meccanismo dei nomi

- ❖ Si può anche inframezzare un gruppo ad un altro:

```
glInitNames();  
glPushName(1);  
// disegna parte del primo gruppo di primitive  
glPushName(2);  
// disegna il secondo gruppo di primitive  
glPopName();  
// continua con il primo gruppo di primitive
```

9

Il meccanismo di selezione

I passi fondamentali della selezione sono:

- ❖ Specificare un array di *unsigned integer* nel quale restituire i nomi degli oggetti selezionati
 - ❖ `glSelectBuffer(<size>, <buffer>)`
- ❖ Portare il sistema in modalità selezione
 - ❖ `glRenderMode(GL_SELECT)` (usando poi lo stesso comando con parametro `GL_RENDER` si torna in modalità di rendering)
- ❖ Specificare la finestrella di selezione (intorno del cursore del mouse)
 - ❖ quest'ultima operazione è piuttosto complessa e dipende dal modo in cui OpenGL gestisce la selezione

10

Il meccanismo di selezione

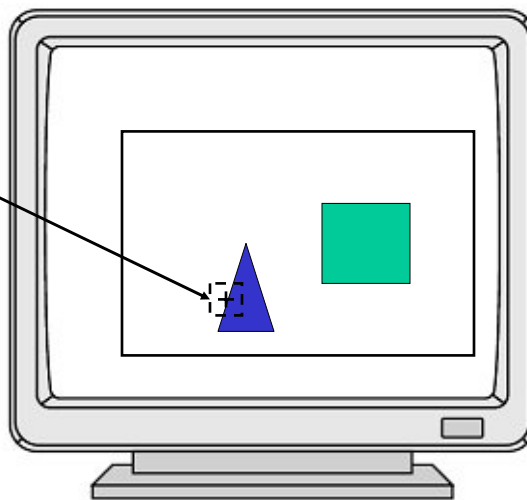
- ❖ OpenGL seleziona le primitive presenti nella regione di pick eseguendo un processo in tutto e per tutto analogo al rendering delle primitive stesse
- ❖ L'unica differenza è che, anziché produrre il risultato del rendering sul framebuffer, fornisce un elenco dei nomi delle primitive "visualizzate"
- ❖ Il volume di vista che si considera per questa operazione è limitato alla porzione che si trova "dietro" la finestrella di visualizzazione

11

Il meccanismo di selezione

Il view volume usato per il picking viene limitato a questa porzione:

- ❖ il triangolo blu è dentro il view volume
- ❖ il rettangolo verde no



12

Specificare l'area di pick

La finestra di pick si specifica con un comando apposito:

- ❖ `gluPickMatrix(x,y,width,height,viewport)`
- ❖ `x,y` sono le coordinate del cursore
- ❖ `width,height` sono le dimensioni della finestrella di picking (estensione intorno al cursore)
- ❖ `viewport` è un array di 4 elementi che indica i limiti della viewport corrente. Si può ottenere tramite:
 - ❖ `glGetIntegerv(GL_VIEWPORT, GLint *viewport);`

13

Specificare l'area di pick

- ❖ Il comando `gluPickMatrix` in realtà definisce una matrice di trasformazione che deve essere accumulata sulla matrice di proiezione per "ritagliare" la regione che ci interessa
- ❖ Bisogna usarlo:
 - ❖ avendo attivato la matrice `GL_PROJECT`
 - ❖ subito prima di definire la trasformazione di proiezione usata in visualizzazione
- ❖ In questo modo la matrice di proiezione viene cambiata in modo da eliminare (mediante clip) tutte le primitive che cadono fuori dell'area di picking

14

Come si fa in pratica...

- ❖ Ammettiamo di avere una procedura **DrawScene ()** che disegna la scena:
 - ❖ Specifica trasformazioni di modellazione, attributi e primitive
 - ❖ Usa il meccanismo dei nomi
 - ❖ È stata usata per visualizzare la scena corrente secondo una determinata trasformazione di vista
- ❖ Ora vogliamo fare picking sulla scena visualizzata

15

...Come si fa in pratica...

- ❖ Si ottengono da GLUT le coordinte **x,y** del mouse (ricordarsi di riportarle al sistema di viewport)
- ❖ Si specifica il select buffer:
 - ❖ **glSelectBuffer (.....) ;**
- ❖ Si porta il sistema in modalità di selezione
 - ❖ **glRenderMode (GL_SELECT) ;**
- ❖ Si seleziona e si salva la matrice di proiezione
 - ❖ **glMatrixMode (GL_PROJECTION) ;**
glPushMatrix () ;

16

...Come si fa in pratica...

- ❖ Si specifica la matrice di pick
 - ❖ `glLoadIdentity(); gluPickMatrix(.....);`
- ❖ Si specificano le stesse trasformazioni di vista usate nel rendering ma *senza usare la `glLoadIdentity` prima di specificare la matrice di proiezione*
- ❖ In questo modo, la trasformazione di vista corrente è la stessa usata prima, ma modificata a posteriori dalla matrice di pick
- ❖ Si richiama la `DrawScene()`;
 - ❖ In questo modo non si visualizza niente, ma i nomi delle primitive che cadono nella finestra di pick vengono scritti nel select buffer

17

...Come si fa in pratica

Per finire:

- ❖ Si ripristina la matrice di proiezione:
 - ❖ `glMatrixMode(GL_PROJECT);`
`glPopMatrix();`
- ❖ Si ripristina la modalità di rendering:
 - ❖ `glRenderMode(GL_RENDER);`
- ❖ Il select buffer può essere a questo punto utilizzato dal programma per accedere agli oggetti selezionati
- ❖ Di solito c'è una corrispondenza tra il nome di un oggetto e la struttura dati del programma che riguarda tale oggetto (cfr. laboratorio)

18

Riferimenti per la lezione 9

❖ Manuale OpenGL – Capitolo 12