

## Afișarea textelor în OpenGL

În OpenGL nu există suport pentru extragerea textelor.

Un Bitmap este o matrice cu valori 0 și 1 ce se poate folosi pentru desenarea de caractere.

Pentru a preciza o poziție curentă de rasterizare (afișare, extragere) se poate folosi comanda:

**glRasterPos{2 | 3 | 4}{s | i | f | d}{(x, y [,z [,w]]) | v}**

Coordonatele curente a poziției de rasterizare sunt transformate în coordonate ecran.

Desenarea unui bitmap se poate face prin comanda:

**glBitmap(width, height, x0, y0, xi, yi, bitmap)**

unde:

**width, height** - sunt dimensiunile bitmap-ului,

**x0, y0** - originea pentru bitmap, care se mută în poziția curentă de rasterizare,

**xi, yi** - valori ce se adaugă la poziția de rasterizare după copiere (se determină următoarea poziție curentă),

**bitmap** - zona cu bitmap-ul (de tip **ByteBuffer**)

Culoarea pentru rasterizare se precizează prin **glColor\*** înainte de apelul comenzii **glRasterPos**.

Exemplu de font precizat printr-o matrice 8x13, unde ultima coloană precizează valoarea hexa a liniei:

			1	1									<b>18</b>
		1	1	1	1								<b>3c</b>
	1	1				1	1						<b>66</b>
1	1						1	1					<b>C3</b>
1	1							1	1				<b>C3</b>
1	1							1	1				<b>C3</b>
1	1	1	1	1	1	1	1	1	1				<b>FF</b>
1	1							1	1				<b>C3</b>
1	1							1	1				<b>C3</b>
1	1							1	1				<b>C3</b>
1	1							1	1				<b>C3</b>
													<b>00</b>
													<b>00</b>

## Utilizare glut

**glRasterPos2f (x,y)**

**glutBitmapString(int font, String string)**

unde primul argument precizează un font, prin una din constantele:

- `BITMAP_8_BY_13`
- `BITMAP_9_BY_15`
- `BITMAP_HELVETICA_10`
- `BITMAP_HELVETICA_12`
- `BITMAP_HELVETICA_18`
- `BITMAP_TIMES_ROMAN_10`
- `BITMAP_TIMES_ROMAN_24`

iar al doilea argument conține șirul care se va extrage.

## Utilizarea clasei `TextRenderer`

Pachetul: `com.sun.opengl.util.j2d.TextRenderer`

Pași:

### 1. Crearea unui obiect `TextRenderer`:

```
TextRenderer text
text = new TextRenderer(new Font("SansSerif", Font.BOLD, 36));
```

### 2. Extragerea (desenarea) unui text în modul următor:

```
text.beginRendering(gld.getWidth(), gld.getHeight());
// stabilire culori, extragere text
text.setColor(1.0f, 0.0f, 0.0f, 1.0f);
text.draw(sir, x, y);
.....
text.endRendering();
```

### 3. Extragerea (desenarea) unui text3D în modul următor:

```
text.begin3DRendering(gld.getWidth(), gld.getHeight());
// stabilire culori, extragere text
text.setColor(1.0f, 0.0f, 0.0f, 1.0f);
text.draw3D(sir, x, y, z, factor_size_font);
.....
text.endRendering();
```