

# Computação Gráfica

Guilherme Henrique de Souza Nakahata

Universidade Estadual do Paraná - Unespar

19 de Junho de 2024

- **Objetos Primitivos:**

- Formas geométricas básicas: Cubos, Esferas, Cilindros e Cones;
- Fundamentais na construção de cenas em computação gráfica;
- Podem ser transformados;
- Combinados para criar formas mais complexas.

- **Posicionamento:**

- Refere-se à localização e orientação dos objetos primitivos;
- Espaço tridimensional;
- Ao posicionar os objetos, é possível representar objetos específicos e criar composições;

- **Hierarquia:**

- Envolve a organização dos objetos em uma estrutura hierárquica;
- Onde objetos podem ser pais ou filhos uns dos outros;
- Essa estrutura permite criar animações e interações complexas.

- **Agrupamento:**

- Técnica de tratar vários objetos como uma única entidade;
- Permite aplicar transformações e atributos em conjunto;
- O agrupamento facilita o gerenciamento e a manipulação de objetos relacionados na cena.

- **Enquadramento:**

- Refere-se à seleção e disposição dos elementos visuais dentro da cena;
- É o processo de posicionar os objetos em relação à câmera e ao campo de visão;
- Destacar elementos importantes, criar equilíbrio visual e guiar a atenção do espectador para pontos-chave da cena.

- A composição de uma cena é a formação e posicionamento de objetos primitivos de tal forma que representem um objeto.

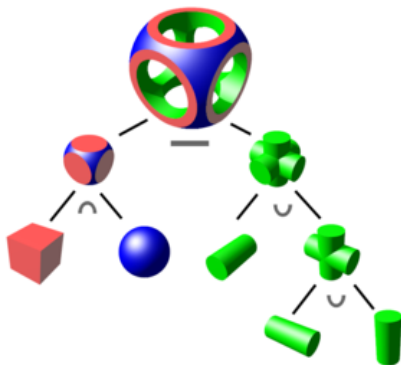


- As transformações e deformações aplicadas à objetos primitivos geram uma composição de objetos, que por sua vez forma uma cena.



- Criar objetos tridimensionais complexos a partir de formas geométricas básicas simples;
  - Cubos;
  - Esferas;
  - Cilindros;
  - Cones;
- É um método que permite a construção de modelos sólidos através da combinação de operações booleanas;
- As três principais operações booleanas:
  - **União:** combinação de dois ou mais sólidos para criar um único sólido que engloba todos eles.
  - **Interseção:** criação de um sólido que representa a área comum entre dois ou mais sólidos.
  - **Diferença:** remoção de uma parte de um sólido usando outro sólido como "ferramenta" de corte.

# Constructive Solid Geometry - CSG



- `glutWireSphere;`
- `glutWireDodecahedron;`
- `glutSolidSphere;`
- `glutSolidDodecahedron;`
- `glutWireCone;`
- `glutWireTeapot;`
- `glutSolidCone;`
- `glutSolidTeapot;`
- `glutWireCube;`
- `glutWireOctahedron;`
- `glutSolidCube;`
- `glutSolidOctahedron;`
- `glutWireTorus;`
- `glutWireTetrahedron;`
- `glutSolidTorus;`
- `glutSolidTetrahedron.`





`glutWireTetrahedron()`



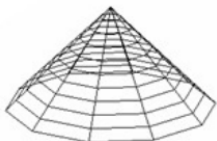
`glutWireDodecahedron()`



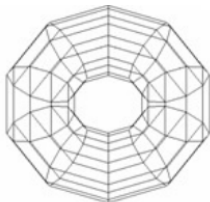
`glutWireOctahedron()`



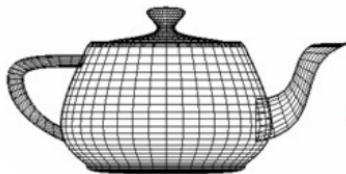
`glutWireIcosahedron()`



glutWireCone()



glutWireTorus()



glutWireTeapot()

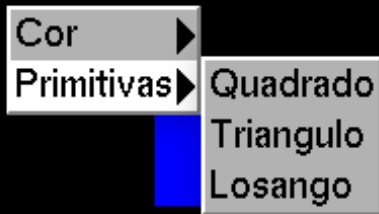
- Bule de Newell;
- Objeto referência;
- Modelo Matemático;
- Sólido;
- Cilíndrico;
- Parcialmente convexo.

- Complexidade;
- Demonstração de algoritmos;
- Benchmark;
- Legado.



- Movimentação do mouse dentro da janela;
- Enquanto não estiver clicando em nenhum dos botões;
- `glutPassiveMotionFunc`;
- Recebe coordenadas X e Y.

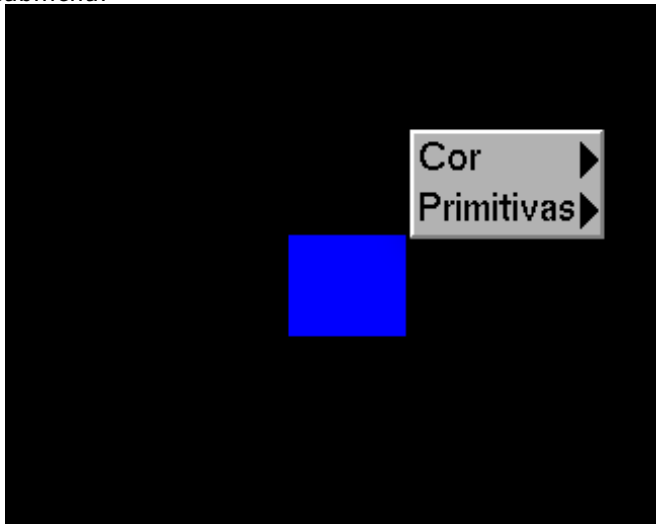
- **glutCreateMenu:** Cria um novo menu pop-up. Parâmetros de entrada (int value), onde value corresponde ao valor que identifica a entrada do menu que foi selecionada.



- **glutAddMenuEntry**: Adiciona uma entrada no final do menu corrente. Os parâmetros de entrada desta função são o nome que será exibido como uma entrada do menu, e o valor que será passado para a função caso esta entrada seja selecionada.



- **glutAddSubMenu:** Adiciona um submenu no final do menu corrente. Os parâmetros de entrada desta função são o nome que será exibido como uma entrada do menu, e o identificador do submenu.





- **glutAttachMenu:** Função que relaciona um botão do mouse com o identificador do menu. O parâmetro de entrada desta função é o identificador do botão do mouse (GLUT\_LEFT\_BUTTON, GLUT\_MIDDLE\_BUTTON ou GLUT\_RIGHT\_BUTTON).

```
void CriaMenu()
{
    int menu,submenu1,submenu2;

    submenu1 = glutCreateMenu(MenuCor);
    glutAddMenuEntry("Vermelho",0);
    glutAddMenuEntry("Verde",1);
    glutAddMenuEntry("Azul",2);

    submenu2 = glutCreateMenu(MenuPrimitiva);
    glutAddMenuEntry("Quadrado",0);
    glutAddMenuEntry("Triangulo",1);
    glutAddMenuEntry("Losango",2);

    menu = glutCreateMenu(MenuPrincipal);
    glutAddSubMenu("Cor",submenu1);
    glutAddSubMenu("Primitivas",submenu2);

    glutAttachMenu(GLUT_RIGHT_BUTTON);
}
```

```
void MenuCor(int op)
{
    switch(op) {
        case 0:
            r = 1.0f;
            g = 0.0f;
            b = 0.0f;
            break;

        case 1:
            r = 0.0f;
            g = 1.0f;
            b = 0.0f;
            break;

        case 2:
            r = 0.0f;
            g = 0.0f;
            b = 1.0f;
            break;
    }
    glutPostRedisplay();
}
```

```
void MenuPrimitiva(int op)
{
    switch(op) {
        case 0:
            primitiva = QUADRADO;
            break;
        case 1:
            primitiva = TRIANGULO;
            break;
        case 2:
            primitiva = LOSANGO;
            break;
    }
    glutPostRedisplay();
}

void MenuPrincipal(int op)
{
}
```

- Movimentação do mouse dentro da janela;
- Enquanto não estiver clicando em nenhum dos botões;
- `glutPassiveMotionFunc`;
- Recebe coordenadas X e Y.

# Obrigado! Dúvidas?

Guilherme Henrique de Souza Nakahata

[guilhermenakahata@gmail.com](mailto:guilhermenakahata@gmail.com)

<https://github.com/GuilhermeNakahata/UNESPAR-2024>