

## SEGUNDA LISTA DE EXERCÍCIOS DA TERCEIRA UNIDADE

### COMPUTAÇÃO GRÁFICA

TERCEIRO ANO - UEMS

PROFESSORA MERCEDES GONZALES MÁRQUEZ

#### Exercício 1.

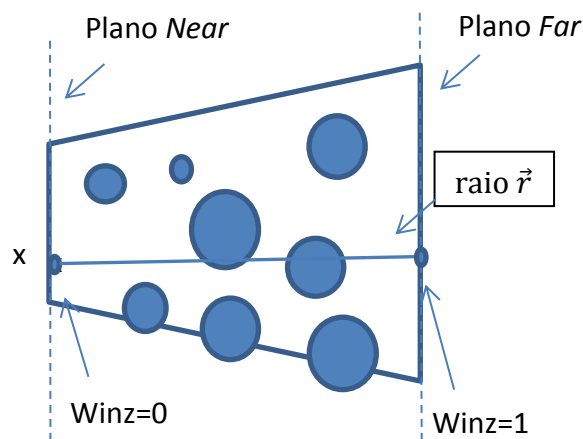
Aprimore o programa canvas.cpp que você apresentou na prova PP1 tal que as figuras na área de desenho (ícones), possam ser clicadas e arrastadas.

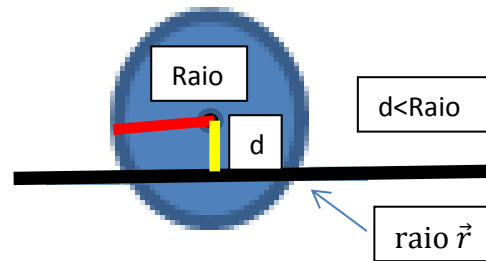
#### Exercício 2.

O comando `gluUnProject` do OPENGL permite obtermos, a partir de um clique do mouse  $(x,y)$ , os pontos extremos de um raio no espaço 3D, cuja projeção coincide com o pixel  $(x,y)$ . Faça um programa em OpenGL e C/C++ que, a partir do clique do mouse, selecione esferas no espaço 3D.

Siga o seguinte processo:

- 1) Clique na tela  $(x,y)$
- 2) Determine o raio  $\vec{r}$  no espaço que corresponde ao clique  $(x,y)$  usando o comando `gluUnProject`.
- 3) Realize a seleção através da determinação da colisão entre o raio  $\vec{r}$  e as esferas. O teste de colisão será o seguinte: calcule a distância  $d$  do centro de cada esfera  $R$  até o raio  $\vec{r}$  usando a distância ponto-reta. Se esta distância  $d$  for menor que o raio  $R$  da esfera, então a esfera passou o teste de colisão e será, por tanto, selecionada.





Material de auxílio:

<https://www.bfilipek.com/2012/06/select-mouse-opengl.html#code> **Select + Mouse + OpenGL using Mouse Ray.**

### **Exercício 3.**

O programa ballAndTorus.cpp do capítulo 4 tem um torus e uma bola azul, acrescente uma bola vermelha e verde tal que as três bolas estejam a uma distância de 120 graus uma da outra e sigam um caminho de rotação e revolução similar uma depois da outra. Adicione a funcionalidade de “pick” qualquer um dos quatro objetos, de forma semelhante ao que é feito no programa ballAndTorusPicking.cpp.

### **Exercício 4.**

Usando os recursos de “picking” crie um game do tipo shoot-em-up.