

Relatório Trabalho final Fundamentos de Computação gráfica

Prof. Eduardo Gastal

Eduardo Bassani e Bruno Loureiro

Descrição dos requisitos na aplicação.

A aplicação está usando muito pouco do código pronto, mas não excluimos o código, então há várias funções fornecidas para as práticas que não estão sendo usadas, mas utilizamos boa parte do código feito nas práticas.

Não utilizamos um controle de tempo para a movimentação dos objetos como foi visto na aula, por falta de tempo, assim se a aplicação rodar em um computador lento não garantimos a movimentação fluida, atualmente rodamos em uma máquina virtual Ubuntu com 2gb ram e 4 cores.

O objetivo da aplicação é percorrer o labirinto no menor tempo possível, desviando dos obstáculos, e não encostando nas paredes, que somem com o tempo, tornando assim o jogo em um jogo de memória e raciocínio.

Utilizamos muito as matrizes de Translação e Rotação, visto que nos controlamos uma esfera, que precisa rodar no chão, para isso passamos o vetor de rotação para as coordenadas do modelo da esfera e assim a rotamos.

As teclas WASD servem para controlar o modelo, o mouse controla a câmera, ESPACE faz o modelo pular e E muda a textura do chão para debug. O resto dos comandos são os mesmos do Laboratório4.

Não conseguimos incluir o cow.obj, sendo que usamos o mesmo código para o bunny.obj e funcionou.

A câmera é uma mistura de câmera livre e look-at, já que ela acompanha o modelo todo o tempo(com translações) e está sempre olhando para ele.

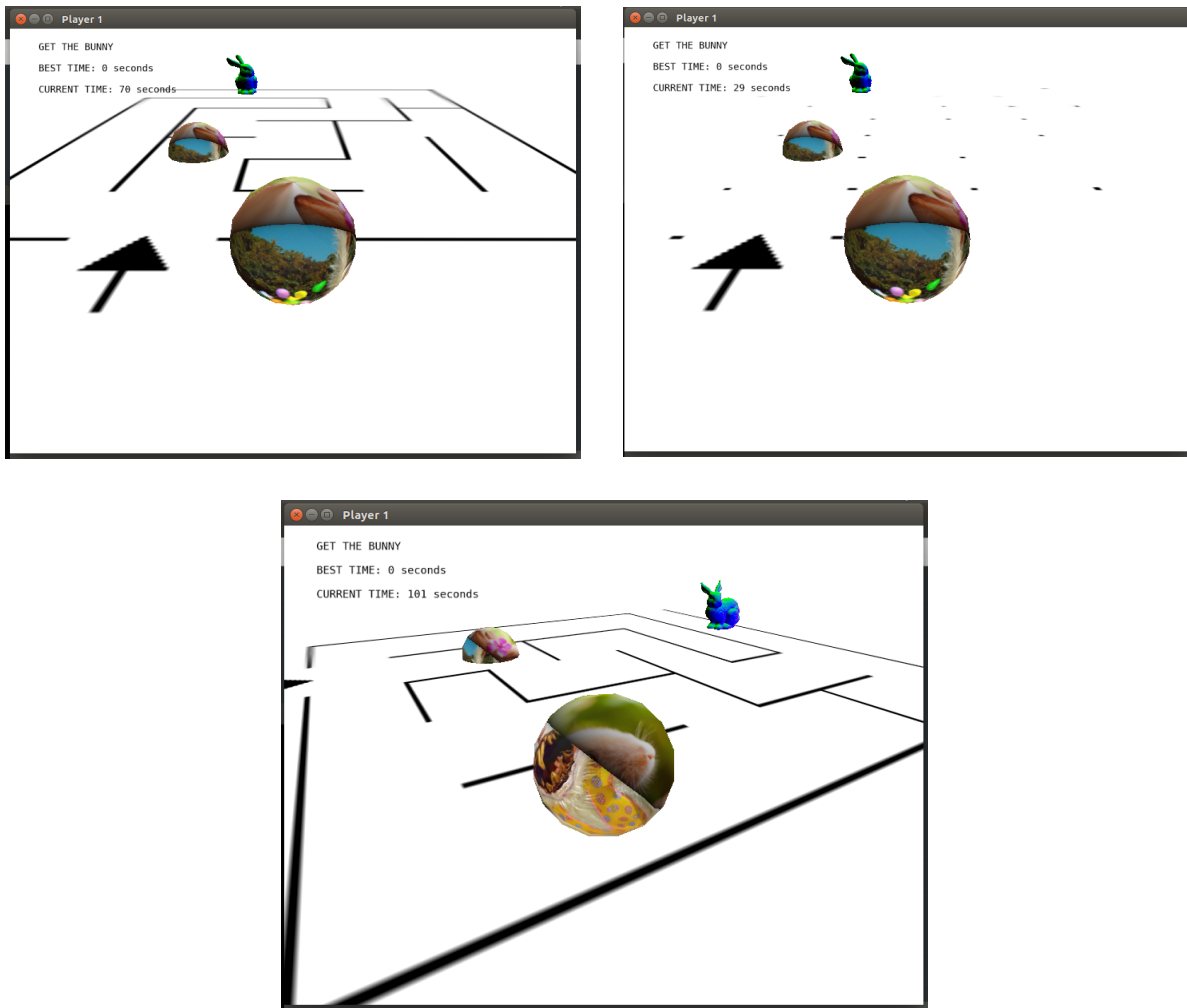
O objeto esfera está sendo copiado duas vezes, a primeira é o modelo que controlamos e a segunda é um obstáculo que se encostado te joga pra longe.

Há teste de intersecção entre esfera-esfera, esfera-plano, esfera-parede, espera-coelho.

O código de iluminação é basicamente o mesmo do laboratório 4, então estamos usando os modelos de iluminação implementados nele.

Todos os objetos são definidos com texturas.

Imagens.



Manual.

Deve-se chegar ao coelho para diminuir o tempo que levamos para chegar nele, desviando dos obstáculos e não encostando nas paredes, as paredes somem a maior parte do tempo, só ficando visíveis durante 1 segundo a cada 10 segundos, então deve-se memorizá-las.

WASD comandam o modelo e ESPACE faz o modelo pular, pode-se usar o mouse para controlar a visualização.

Compilação.

O código foi feito no Ubuntu, para compilar e rodar basta executar o comando “make && make run” na pasta do projeto.