

MAC0420/5744 – Introdução à Computação Gráfica

1o Semestre de 2011

Trecos - Um jogo 2D

Exercício-Programa Um (EP1)

Data de entrega: sábado, 09/04/2011, até 23:55h

Entrega: até 23:55hs de 9 de abril de 2011.

Grupos: O projeto é individual.

Objetivos: Familiarizar você com as funcionalidades básicas do OpenGL, GLUT, e programas baseados em eventos, em particular animações. Você deverá implementar um vídeo game bidimensional (2D), que chamaremos de **Trecos**. Você pode fazer ligeiras modificações na descrição abaixo para melhorar a interface com o usuário, mas você deve me comunicar qualquer modificação por email e esperar minha autorização.

O que entregar: Os fontes e o makefile para o seu programa no formato tgz. Você deve se assegurar que seu programa compile e rode na rede Linux do IME.

Descrição: Podemos pensar nesse jogo como sendo passado em uma piscina, embora a parte gráfica seja desenhada em um plano (x,y). A piscina é povoada com objetos chamados trecos. Cada treco é desenhado como um quadrado colorido no plano (x,y) - mas você pode desenhar como uma outra forma qualquer. Os trecos se movem dentro de um grande quadrado com velocidade constante. Quando um treco bate no canto do quadrado, ele é rebatido mas mantém sua velocidade. Ou seja, se o treco bate em uma parede vertical, sua velocidade y se mantém e a componente x é invertida. Inicialmente os trecos começam com posições e velocidades aleatórias.

Cada treco flutua em um certo nível na piscina, e eles não mudam de nível. Você pode considerar a profundidade da piscina como o eixo z. Vamos considerar uma piscina com 5 níveis de profundidade, ou seja, $z = 0, 1, 2, 3, 4$, sendo que zero está próximo a superfície, e 4 próximo ao fundo da piscina. Cada nível contém inicialmente 4 trecos. Os trecos dos níveis 0, 1 e 2 são bons (desenhados em amarelo) pois eles limpam a piscina, e os trecos dos níveis 3 e 4 só a poluem (são trecos maus, desenhados em violeta). Vamos considerar também que existe o nível 5, o fundo, mas sem nenhum treco.

O objetivo do jogo é "matar" todos os trecos ruins (você ganha pontos) sem "matar" os trecos bons (você perde pontos). Trecos no mesmo nível possuem a mesma cor. Você deve desenhar os trecos mais profundos com uma cor mais escura, para dar a noção de profundidade.

Você mata um treco lançando bombas na piscina. Uma bomba é lançada clicando o botão esquerdo do mouse em algum lugar da piscina. A bomba começa no nível 0 e desce lentamente para os demais níveis, até atingir o fundo, quando a bomba desaparece. Uma bomba é desenhada como um quadrado de menor tamanho. Para indicar a sua profundidade, a cor da bomba muda para corresponder à cor dos trecos no mesmo nível. Quando uma bomba no nível i atinge um treco ele é ferido (se torna vermelho). Um segundo depois o treco ferido morre (ele desaparece). A bomba porém continua a descer até atingir o fundo (mas você pode considerar que a bomba some ao matar um treco).

O jogo termina quando todos os trecos ruins são eliminados, ou quando o usuário aperta a tecla 'q' (quit). O jogo pode ser interrompido (pausa) pela tecla 'p', e mostrado passo-a-passo pelo botão direito do mouse (para facilitar a depuração do programa). A seguir estão os eventos que o seu programa deve controlar:

Eventos de entrada:

- **Teclado:** a tecla 'q' termina o jogo, e a tecla 'p' interrompe-o, 'congelando' o jogo. Quando a tecla 'p' é pressionada novamente, o jogo continua. (como teste, tente modificar o tamanho da tela durante uma

pausa).

- **Botão esquerdo do mouse:** lança uma bomba sobre a posição corrente do mouse. Não há limite no número de bombas ativas, a piscina toda pode ficar cheia de bombas. As bombas também nunca acabam.
- **Botão direito do mouse:** quando o botão direito do mouse é pressionado, o jogo é interrompido imediatamente se ele estiver rodando. Caso ele já esteja interrompido, o jogo roda um passo (você pode definir o tamanho do seu passo) e imprime informações relevantes sobre o estado do jogo. O jogo continua normalmente ao pressionar a tecla 'p'.
- **Reformatar a janela:** ao mudar o tamanho da janela, o seu programa deve fazer algo "adequado". Em particular, deve manter um aspecto visual "agradável", se comportar da mesma forma, e manter o mesmo "nível de dificuldade" (por exemplo, se a janela é aumentada, o tamanho e as velocidades devem ser aumentadas proporcionalmente, não sendo necessário que eles se mantenham quadrados).

Eu vou deixar para a sua criatividade definir a maneira exata de como o seu jogo deve se comportar (lembre-se que o objetivo é aprender OpenGL e GLUT desenvolvendo um jogo 2D). Os tamanhos e velocidades vão ser especificadas em quantidades virtuais chamadas *lembas*, que você pode ajustar para redimensionar a sua janela. Os pontos do jogador assim como outras informações relevantes devem ser mostradas na tela. Abaixo estão alguns parâmetros que você pode utilizar (e modificar se quiser):

- Tamanho de uma lemba: 400 pixels
- Tamanho inicial da janela (em lembas) 1x1
- Comprimento de um treco: 0.10 lemba
- Velocidade do treco: Uniformemente distribuido entre 0.0 a 0.25 lembas por segundo, para cada coordenada (dimensão x ou y).
- Tamanho da bomba: 0.04 lembas.
- Velocidade da bomba: 1 nível por segundo
- Número de trecos: 5 níveis com 4 trecos por nível
- Pontuação: nível 0: -4000; nível 1: -2000; nível 1: -1000; nível 3: +1000; nível 4: +2000;

Dicas para melhorar o seu projeto:

- **Beleza:** melhore os desenhos dos trecos e das bombas para torná-los mais fáceis de perceber (com adicionar uma borda escura).
- **Dicas de profundidade:** forneça melhores informações visuais sobre a profundidade dos trecos. Para as bombas, ao invés de simplesmente mudar de cor entre níveis, faça com que as bombas mudem de cor gradualmente, ou mostre um número correspondente ao nível.
- **Opções:** forneça um menu que permita ao usuário alterar algumas configurações do jogo, e um help que descreva as instruções.
- **Efeitos especiais:** verifique a colisão entre trecos no mesmo nível, adicione rotação aos trecos, adicione sombras, anime o processo de um treco sendo atingido por uma bomba, etc.

Importante: Comece seu trabalho logo e não esqueça de documentá-lo. Em particular, seu programa precisa conter um cabeçalho com seu nome, nusp e uma breve descrição do programa.