

MAC 420/5744 – Introdução à Computação Gráfica

Prof. Marcel Parolin Jackowski

BCC - IME/USP – Primeiro Semestre de 2014

Segundo Exercício-Programa

Data de entrega: até 08/junho/2014

Data de apresentação: 11/junho/2014

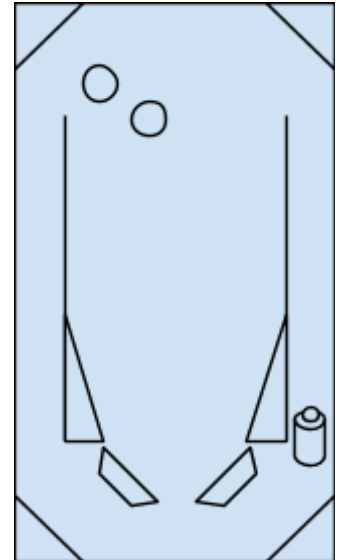
Pinball 3D em WebGL

Neste exercício-programa colocareis em prática os seguintes conhecimentos adquiridos em CG até o momento: modelagem e representação geométrica, transformações lineares, projeções, iluminação, sombras, textura e física clássica. Desenvolvereis um programa (possivelmente contendo vários módulos) em Javascript e WebGL que simulará um jogo de *Pinball* 3D com obstáculos e sistemas de colisão. O objetivo é reproduzir uma máquina de Pinball com palhetas, esferas e obstáculos, além de componentes interativos para permitir um usuário jogar a esfera, de modo a ganhar pontos e evitar perdê-la. Deve-se efetuar a animação das jogadas até o fim da partida. O funcionamento do Pinball está descrito em <http://pt.wikipedia.org/wiki/Pinball>. Podes jogar neste site <http://ie.microsoft.com/testdrive/Graphics/CanvasPinball/default.html>

Outra inspiração é o site da FuturePinball: <http://www.futurepinball.com>

Máquina de Pinball

Devereis modelar uma máquina de Pinball com elementos que sirvam de obstáculos para a esfera. A figura do lado direito mostra um layout de exemplo (senti-vos livre para inserirdes quantos obstáculos com os funcionamentos mais variados a desejardes). Inserir texturas na máquina para aumentar seu grau de realismo. Iluminação também é importante. Inserir uma fonte de luz na máquina de forma que apareça características difusas e especulares nos objetos. Colocar uma iluminação ambiente para simular a interação entre os elementos (de forma com que as faces que não receberem luz não estejam totalmente escuras). Podereis criar as faces diretamente no código ou criar objetos em um programa de modelagem (Blender, Maya, 3D Studio Max...) e exportar num formato OBJ. Você pode utilizar o *parser* criado no desenvolvimento do EP1 (modificando-o para acomodar as novas propriedades - coordenadas de textura e normais). A máquina deverá no mínimo conter: dispositivo de lançamento da esfera (pode ser uma mola, um cilindro, ou qualquer outro formato), paletas que serão controladas pelo usuário (via teclado) para manter a esfera em jogo, obstáculos (paredes, discos, cantos, inclinações, outras molas, outras palhetas automáticas...) que personalizará o ambiente e dará o dinamismo do jogo.



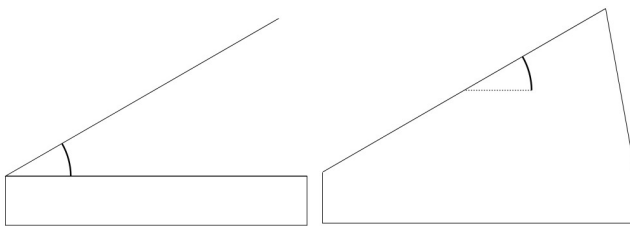
A superfície da mesa da máquina deverá estar inclinada a X graus (número obtido do arquivo de configuração do mapa). Mesmo assim, senti-vos à vontade para posicionar e orientar a câmera como desejardes.

Dinâmica do Jogo

Ao entrar na página, o jogador deve visualizar a máquina juntamente com a interface de comandos (se preferirdes, podeis criar um *splash screen* antes dessa tela). A interface deve mostrar a pontuação do jogador, controles para ativar ou desativar a iluminação, ativar ou desativar o som (vide seção Áudio),

pausar/continuar e reiniciar o jogo, e mudar a personalização da máquina. No seu estado inicial, a máquina está com uma esfera no lançador e com pontuação zerada. Segurando a tecla de **Espaço**, o lançador começa a ganhar força (se distanciando da sua posição original). Ao soltar a tecla de **Espaço**, a esfera é lançada para cima. Fazei com que a esfera vá para a parte principal da máquina (pode ser com algum componente curvo ou retilíneo). As teclas **Z** e **X** permitem rotacionar as palhetas para impulsionar a esfera, caso ela esteja descendo na mesa. Enquanto a esfera estiver percorrendo a mesa, ela deve estar rolando no sentido perpendicular ao sentido do movimento. A rotação da esfera deve estar de acordo com sua velocidade naquele momento (convertendo velocidade espacial em angular). Modelar a desaceleração pela força de atrito não é obrigatório, mas é obrigatória a (des)aceleração da esfera devido à inclinação da mesa, à força da gravidade, a massa da esfera (pode ser uma constante) e colisões. Na maioria das vezes a esfera irá se chocar com outros elementos da máquina. Aplicai uma colisão inelástica (com perda de energia), calculando corretamente o novo sentido da esfera. Adicionai pontos para o jogador em colisões com alguns dos tipos de obstáculo. O jogador tem 3 chances no jogo (mostrai isso na interface). Se a esfera cair no fundo da mesa, retirai-la e colocai no lançador, retirando uma chance disponível. Terminadas as chances, finalizai o jogo e destaque a pontuação.

Inclinação



Permiti que o jogador mude a inclinação da mesa usando o teclado (aumentando e reduzindo a angulação). O efeito visual fica ao vosso critério (distorcendo ou não as faces, rotacionando o conjunto todo...). O eixo de rotação é o eixo X, porém no canto inferior, no centro ou em qualquer ponto da mesa.

Obstáculos

Em um dos tipos de obstáculos, fazei com que, ao colidir com a bola, novas bolas (mínimo de 3) são criadas ao redor do obstáculos, dispersando-os com a mesma energia da bola original. Desativai esta funcionalidade enquanto houver mais de uma bola na mesa. Ativai quando apenas houver uma bola. Fazei com que os obstáculos mudem algum aspecto físico da bolinha, especialmente sua energia e, conseqüentemente, sua velocidade, para menos ou para mais.

Iluminação

Permiti ao jogador escolher a iluminação através da interface: desativar iluminação (cores e texturas são exibidas sem efeitos), cores da luz ambiente, difusa e especular, além da potência da luz. Posicionai a luz como desejar, além de definir seu tipo (omni, direcional, spot, area...). Criai um skybox (seja um céu, uma sala...) em volta do ambiente, que será influenciado pelo Modo Cheat (caso ativado).

Bônus – Sombras

Criai sombras usando qualquer uma das técnicas (projeção, shadow mapping) para criar hard e/ou soft shadows. Inicialmente as sombras devem aparecer na mesa, todavia senti-vos livre para aplicar sombras entre objetos.

Bônus – Áudio

Permitai sons para eventos como: lançamento da esfera, colisão entre objetos, movimento da palheta, a esfera caindo no buraco, entre outros. Se desejardes, podereis também adicionar uma música de ambiente.

Para saberdes mais sobre a API de Áudio HTML5: <http://www.html5rocks.com/en/tutorials/webaudio/intro/>

Alguns arquivos de áudio de exemplos estão disponibilizados em conjunto com este documento.

Entrega e Demonstração

A entrega do produto deve ser feita no Paca, compactado, com vossos nomes, no dia 8/06, prorrogável até 15 minutos antes da efetiva demonstração. No dia 11/06, estareis realizando a demonstração do jogo. Mostrai todo o potencial e diversão que o jogo proporciona. Se desejardes, mostrai também o que foi usado e as dificuldades de se trabalhar com um projeto complexo como este. Adicionai vosso projeto nas páginas pessoais.

Restrições

- Evitar o uso de bibliotecas gráficas, como Three.js, GLGE entre outros;
- Evitar o uso de bibliotecas de física;
- Podeis usar bibliotecas de manipulação de matrizes, textos, quaternions, carregamentos de objetos (feitos por vós) e carregamento de imagens (apenas o carregamento, pois o *upload* das texturas para a GPU devem ser feitas usando os recursos diretos da WebGL);
- Qualquer outra biblioteca deve ser posta sobre questão no fórum;

Ficha de Avaliação

#	Descrição	Pontuação
1	Mesa e obstáculos exibidos corretamente	05
2	Paletas, bolinhas, e lançador exibidos corretamente	05
3	Paletas se movimentam corretamente	03
4	Paletas impulsionam a esfera corretamente	03
5	Obstáculos funcionando corretamente	08
6	Ao perder a esfera, uma chance é gasta e o jogo é recomeçado	03
7	Ao perder 3 chances, o jogo termina.	03
8	Pontuação funcionando corretamente	05
9	Interface correta	08
10	Iluminação correta (alterando tons de textura e cores)	05
11	Iluminação atualizada pelos controles da interface	04
12	Skybox feito corretamente	05
13	Lançador funcionando corretamente	05
14	Esfera se movimenta na velocidade e aceleração corretas	05
15	Esfera se rola corretamente	04
16	Dinâmica do jogo funcionando efetivamente	05
17	Várias bolinhas funcionam corretamente (sist. de partículas > 8fps)	05
18	Inclinação correta (teclado e física)	05
19	Botões pausar/continuar e reiniciar funcionando corretamente	04
20	Projeto compactado, entregue e demonstração feita satisfatoriamente	10
21	Bônus – sombras feitas corretamente	50
22	Bônus – áudio funcionando corretamente	50
Total (Sem Bônus)		100
Total (Apenas Bônus)		100

OBS: Essa pontuação é apenas um percentual. A nota final do EP (que comporá a média final do aluno) será proporcional ao peso máximo dado pelo EP.

OBS₂: Os bônus são independentes da pontuação normal.