	Centro Tecnológico Departamento de Informática	
Disciplina: Computação Gráfica		Código: INF09282 e INF09284
Prof. Thiago Oliveira dos Santos		

Trabalho 2D

1 Introdução

Esse trabalho tem como objetivo avaliar o conhecimento dos alunos em relação computação gráfica interativa 2D.

Para isso, o aluno deverá implementar um programa de luta de boxe. O programa será composto por dois jogadores que lutarão em uma arena. O personagem do jogador será controlado pelo teclado e pelo mouse, e poderá dar socos com ambos os braços. O trabalho deverá ser implementado em C++ (ou C) usando as bibliotecas gráficas OpenGL e GLUT (freeglut).

2 Especificação das Funcionalidades

Ao rodar, o programa deverá ler e interpretar os elementos da arena do arquivo do tipo SVG informado pela linha de comando. A arena será composta por uma série de elementos (ver Figura 1): uma arena representada pelo retângulo azul; um círculo verde representando o personagem do jogador e um círculo vermelho representando o oponente (controlado pelo computador). Um arquivo SVG será fornecido como exemplo juntamente com a descrição do trabalho, porém é responsabilidade do aluno testar outros arquivos com configurações (posições e tamanhos) diferentes para os elementos. A leitura do SVG poderá ser feita utilizando-se um parser para XML. Sugiro utilizar a Tinyxml que é simples e pode ser enviada juntamente com o código para ser compilada.

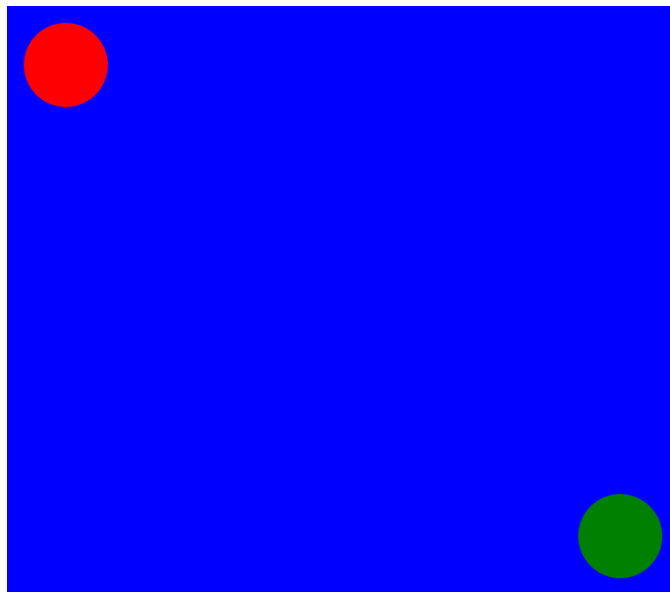



Figura 1: Arquivo SVG representando a arena em azul, jogador em verde e oponente em vermelho.

Desenho

Os elementos lidos do arquivo SVG vão servir somente para desenhar a configuração inicial do jogo e não para desenhar os personagens completos, portanto ele conterá somente os elementos e cores descritos anteriormente. A arena deverá ocupar toda a janela, ou seja, o canto superior esquerdo e o canto inferior direito do retângulo da arena (visto na Figura 1) deverão respectivamente coincidir com o canto superior esquerdo e o canto inferior direito da janela do programa. A janela deverá ter a mesma dimensão (em pixels) do retângulo da arena e pode ser posicionada a gosto na tela. Não é necessário tratar o *resize*.

	Centro Tecnológico Departamento de Informática	
Disciplina: Computação Gráfica		Código: INF09282 e INF09284
Prof. Thiago Oliveira dos Santos		

Os círculos verde e vermelho do xml servirão somente para saber onde e com qual tamanho desenhar respectivamente os personagens do jogador e do oponente (utilizar as cores do círculo para diferenciar os personagens ao desenhá-los). Os círculos representarão a cabeça dos jogadores e estarão sempre dentro da arena. Os outros elementos para compor os jogadores devem escolhidos e desenhados pelo aluno usando uma escala controlada pelo tamanho da cabeça. Por exemplo, se o personagem tiver uma cabeça de raio X e um braço de comprimento Y , um personagem com a cabeça de raio $X/2$ deverá ter um braço de comprimento $Y/2$. Os personagens serão compostos pelos elementos descritos na Figura 2. O nariz indica a frente do personagem. Ao iniciar o jogo os personagens devem estar de frente um para o outro.

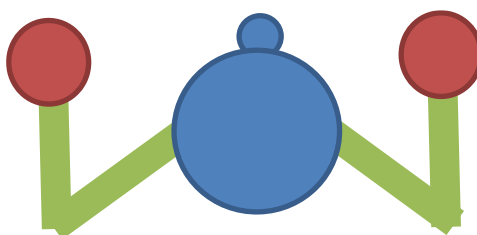


Figura 2: Ilustração do personagem composto por uma cabeça com um nariz (em azul), dois braços (em verde) e duas luvas (em vermelho).

Andar

O controle do movimento de andar será feito com teclas a , s , d e w do teclado. As teclas a e d servirão para girar o corpo do jogador no sentido anti-horário e horário respectivamente. As teclas w e s servirão para mover o jogador para frente (direção do nariz) e para trás respectivamente. A combinação de duas teclas, uma de giro e uma de movimento, deverão causar um movimento curvilíneo, como se estivesse fazendo uma curva.

Colisão

Os jogadores não deverão sair da arena (a cabeça não deve sair) e não deverão ocupar um mesmo espaço. Para não ocupar o mesmo espaço, a colisão deverá ser calculada considerando um círculo imaginário envolvendo o jogador. O círculo imaginário não poderá entrar no espaço da cabeça do outro jogador, conforme representado na Figura 3. Perceba que o círculo imaginário não precisa ser desenhado.

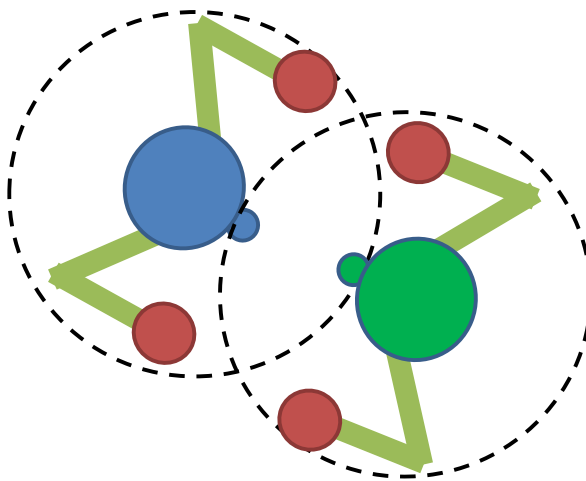



Figura 3: Ilustração da colisão entre dois personagens. O círculo imaginário é representado pelo círculo tracejado que não deve ocupar o espaço do círculo da cabeça do adversário.

	Centro Tecnológico Departamento de Informática	
Disciplina: Computação Gráfica		Código: INF09282 e INF09284
Prof. Thiago Oliveira dos Santos		

Dar Socos

O jogador poderá dar socos com os dois braços, um por vez. O soco será controlado pelo mouse. Para disparar um soco, o usuário deverá clicar o botão esquerdo do mouse e mantê-lo pressionado para controlar os braços. Após o clique, o jogador entrará em situação de soco (i.e., começou a dar um soco) e ao mover o mouse para a direita o braço direito iniciará o movimento de soco para frente. O braço chegará em sua posição máxima após o mouse percorrer uma distância equivalente à metade da largura da janela (considerando a posição inicial como ponto de partida). Após essa distância, o braço deve permanecer na posição máxima mesmo que o mouse continue se afastando com o botão clicado. O braço retornará à posição inicial ao retornar com o mouse para a posição inicial do clique. Só é necessário considerar o movimento horizontal do mouse para disparar o soco. O movimento análogo pode ser feito para o braço esquerdo. Portanto, para dar vários socos na sequência, basta manter o mouse pressionado e movê-lo de um lado para o outro. Ao soltar o botão ou andar com o jogador, o jogador sairá da situação de soco e o braço deverá retornar para a posição inicial mesmo que o movimento não tenha terminado.

O movimento de soco está ilustrado na Figura 4. Durante o movimento de soco, o braço move a junta do ombro e do cotovelo até sair do círculo imaginário de colisão. A proporção de movimento das juntas deve ser calibrada para fazer o movimento visualmente apropriado.

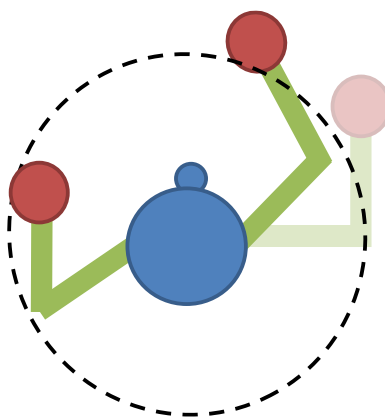


Figura 4: Ilustração do movimento de soco com o braço direito.

Pontuar

Toda vez que uma das luvas colidir com a cabeça do adversário enquanto o usuário estiver em situação de soco (i.e., com o botão pressionado e fazendo o movimento de soco), um ponto deve ser marcado e a cabeça atingida deverá mudar de cor enquanto houver a colisão. Perceba que para contar dois pontos a luva deve colidir com a cabeça, deixar de colidir e depois colidir novamente. O jogo acaba com a vitória de quem atingir 10 pontos. O placar deve ser mostrado no canto inferior esquerdo da tela. Ao finalizar o jogo, deve-se mostrar uma mensagem nela com o vencedor.


Movimento do oponente

O oponente deverá ter algum movimento “aleatório” para atacar o jogador. Ele deverá dar socos e buscar a luta com o jogador, porém respeitando as ações de movimento estabelecidas para andar e para dar socos. O movimento “aleatório” é de livre escolha, mas ele deve buscar a luta e permitir a correção do trabalho (ou seja, visualização das tarefas do trabalho).

Geral

O aluno deverá tratar os movimentos para funcionar de forma equivalente em máquinas diferentes (ou seja, tratar o tempo decorrido entre chamadas *idle*), conforme vista em sala. Ele deverá também utilizar os conceitos de *double buffer* e variável de estado das teclas para interação com teclado (usando o evento KeyUp como visto em aula). A utilização de conceitos de modularização (e.g. usando classes para representar os objetos da cena) facilitará a implementação do trabalho seguinte.

3 Regras Gerais

	Centro Tecnológico Departamento de Informática	
Disciplina: Computação Gráfica		Código: INF09282 e INF09284
Prof. Thiago Oliveira dos Santos		

O trabalho deverá ser feito individualmente. Trabalhos identificados como fraudulentos serão punidos com nota zero. Casos típicos de fraude incluem, mas não se restringem à cópia de trabalhos, ou parte deles, assim como trabalhos feitos por terceiros.

3.1 Entrega do Trabalho

O código deverá ser entregue pelo Google Classroom dentro do prazo definido. Trabalhos entregues após a data estabelecida não serão avaliados.


A entrega do trabalho deverá seguir estritamente as regras a seguir. O não cumprimento **inviabilizará a correção do trabalho** que, por sua vez, receberá nota zero.

- Arquivo zippado (com o nome do autor, ex. FulanoDaSilva.zip) contendo todos os arquivos necessários para a compilação do trabalho;
- **Não enviar** arquivos já compilados, inclusive bibliotecas!
- O arquivo zip deverá necessariamente conter um *makefile* que implemente as seguintes diretivas "make clean" para limpar arquivos já compilados, "make all" para compilar e gerar o executável. O executável deverá ser chamado *trabalhocg*.

Lembre-se que a localização do arquivo da arena será passada via linha de comando e, portanto, não se deve assumir que haverá um arquivo desses na pasta do executável. Seja cuidadoso ao testar o seu programa, isto é, não teste com o arquivo no diretório do programa, pois você pode esquecer de testá-lo em outro lugar posteriormente.

3.2 Apresentação do Trabalho

O trabalho terá 15 minutos para ser apresentado para a turma. A apresentação será feita para a turma no laboratório. As apresentações ocorrerão no horário da aula e em uma data posterior à de entrega, conforme cronograma das aulas. Durante o tempo de apresentação, o aluno deverá mostrar e testar todas as funcionalidades requeridas do trabalho. O trabalho (arquivos) a ser utilizado na apresentação deverá ser o mesmo enviado para o professor, e será fornecido pelo professor na hora da apresentação. A ordem de apresentação será sorteada durante a aula, portanto, todos os alunos devem estar preparados para apresentar o trabalho durante o período de apresentações. Os alunos devem estar preparados para responder possíveis perguntas sobre o trabalho. Prepare-se para fazer a apresentação dentro do seu tempo. **Pontos só serão dados para funcionalidades apresentadas**, isto é, a audiência deverá ser capaz de ver e perceber o resultado produzido pela funcionalidade implementada no jogo. Cabe aos alunos, portanto, criar atalhos (para habilitar e desabilitar funcionalidades, por exemplo, movimento do oponente) no trabalho para facilitar a apresentação das funcionalidades.

	Centro Tecnológico Departamento de Informática	
Disciplina: Computação Gráfica		Código: INF09282 e INF09284
Prof. Thiago Oliveira dos Santos		

3.3 Pontuação

A avaliação do trabalho seguirá a tabela abaixo, com bug sendo descontados caso a caso.

Itens	Sub-Itens	Feito	Observações	Pontos	Nota
Desenho	Arquivo dado			2,0	
Andar	Andar corretamente			2,0	
	Manter as partes do corpo				
Colisão	Não sair da arena			2,0	
	Não invadir o oponente				
	Não ser invadido pelo oponente				
Dar Socos	Movimento			2,0	
	Movimento em posições/ângulos diferentes				
	Socos em sequência				
Pontuar	Colisão/Mudar cor			0,5	
	Contagem escrita			0,5	
Movimento do oponente	Anda corretamente			1,0	
	Da socos corretamente				

4 Erratas

Qualquer alteração nas regras do trabalho será comunicada durante a aula e no portal. É de responsabilidade do aluno frequentar as aulas e se manter atualizado.