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

Trabalho Curto 4

1 Introdução

Esse trabalho tem como objetivo integrar os conceitos dos trabalhos anteriores em um pequeno jogo.

O jogo terá como objetivo destruir todas as bases inimigas sem tomar um tiro ou colidir com os inimigos voadores. O jogo acaba quando você destrói todas as bases inimigas (vencendo), ou quando você morre (perdendo).

O aluno deverá implementar um programa que incremente o trabalho curto 3 (TC3) para adicionar as funcionalidades descritas aqui. Neste trabalho, os componentes da arena serão dinâmicos, ou seja, os inimigos deverão se mover “aleatoriamente” (o tipo de movimento é de livre escolha do aluno, mas seguindo os movimentos básicos definidos para o jogador no TC3, isto é, voar como se fosse o jogador). Além disso, os tiros deverão interagir com os inimigos (ou seja, ao atingi-los, deverão matá-los) e vice-versa (fogo amigo não mata). 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, de um arquivo de configurações (denominado “config.xml”), as configurações necessárias para suas tarefas. O arquivo de configurações deverá estar no formato xml e será fornecido juntamente com a especificação do trabalho. A localização do arquivo “config.xml” será fornecida pela linha de comando ao chamar o programa. Por exemplo, se o arquivo estiver dentro de uma pasta chamada “Test1” localizada na raiz, basta chamar o programa com “./Test1” como argumento (outros exemplos de caminhos possíveis “../Test1/”, “../../Test1/”, etc.). As informações contidas nesse arquivo servirão para ler o arquivo SVG contendo as informações da arena.

Além das tags já especificadas no trabalho curto 2 e 3, o arquivo de configurações deverá conter uma sub-tag específica para descrever parâmetros dos inimigos, denominada <inimigo>. A tag <inimigo> terá, além de um atributo de velocidade do inimigo e velocidade do tiro, similares às tags do jogador (vide TC3), um atributo “freqTiro” para determinar a frequência dos tiros dos adversários, isto é, quantos tiros os adversários darão por instante de tempo. Ele será dado em número de tiros por segundos.


Exemplo do arquivo config.xml

```
<aplicacao>
  <arquivoDaArena>
    <nome>arena</nome>
    <tipo>svg</tipo>
    <caminho>../Test1/</caminho>
  </arquivoDaArena>
  <jogador velTiro="2.0" vel="1.0"></jogador>
  <inimigo freqTiro="0.2" vel="2.0" velTiro="1.0"></inimigo>
</aplicacao>
```

Após ler as informações do arquivo de configurações, o programa deverá carregar os elementos da arena do arquivo do tipo SVG respectivo e colocar um jogador verde ao invés de um círculo verde, jogadores inimigos vermelhos ao invés de círculos vermelhos, além das bases inimigas (estas poderão ser substituídas por algum objeto de interesse, escolhido pelo aluno, que caiba no círculo). A forma dos personagens inimigos será conforme descritos para o jogador no TC3.

Tiro

Todos os inimigos poderão atirar. Os tiros do inimigo deverão matar o jogador quando houver colisão com o tiro. Da mesma forma, o tiro do jogador poderá matar o inimigo. O jogador terá o canhão controlado conforme o TC3, e os inimigos deverão atirar de tempos em tempos (utilizar o parâmetro “freqTiro” do arquivo de configurações). Tiro amigo não mata! Ou seja, os inimigos não podem se matar entre si. O canhão do inimigo não precisa mover.

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

Bases inimigas

A base inimiga só poderá ser destruída por uma bomba. A destruição ocorrerá quando a bomba cair (i.e., após esgotar o tempo de queda definido no TC3) dentro do círculo da base inimiga. A base inimiga deverá sumir ao ser destruída. As bombas só destruirão as bases inimigas, não tendo efeito sobre os inimigos. Os inimigos não atiram bombas.

Mover Inimigos

O movimento dos inimigos deverá seguir as regras do movimento do jogador. Os inimigos não devem colidir entre si, ou seja, eles devem passar por cima uns dos outros. Cada inimigo deverá ficar se movendo aleatoriamente (algoritmo de livre escolha do aluno, mas respeitando as propriedades de movimentos definidas no TC3) e de tempos em tempos atirar (considerar a frequência dada no arquivo de configurações). Exemplos de movimentos, dar voltas em torno do centro da arena; voar em zig zag; de tempos em tempos escolher uma direção e ir, etc. O jogador deverá se colidir normalmente com os inimigos quando ambos e isso finalizará o jogo com uma derrota.

Finalização do Jogo

O jogo deverá conter um placar no canto superior direito indicando o número de bases destruídas até o momento e o número que faltam. No final do jogo, uma mensagem deverá ser impressa no meio da tela dizendo se você ganhou ou perdeu. O jogador ganha se ele destruir todas as bases inimigas, e perde se levar um tiro.

3 Regras Gerais

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 trabalho, ou parte dele, assim como trabalhos feitos por terceiros. Caso seja necessário confirmar o conhecimento do aluno a respeito do código entregue, o professor poderá pedir ao aluno para apresentar o trabalho oralmente em um momento posterior. A nota da apresentação servirá para ponderar a nota obtida no trabalho.

3.1 Entrega do Trabalho

O código deverá ser entregue por email (para: todsantos@inf.ufes.br) dentro do prazo definido no portal do aluno. Trabalhos entregues após a data estabelecida não serão corrigidos.


A entrega do trabalho deverá seguir estritamente as regras a seguir. O não cumprimento acarretará na **não correção do trabalho** e respectivamente na atribuição da nota zero.

- Assunto da mensagem: [CG-2019-2] <tipo do trabalho>. Onde, <tipo do trabalho> pode ser TC1, TC2, TC3 e representa respectivamente trabalho curto 1, 2, 3, etc , ou TF para o trabalho final.
- Anexo da mensagem: 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 diretório 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 config.xml 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 testa-lo em outro lugar posteriormente.

4 Pontuação

O trabalho será pontuado conforme a tabela abaixo. Bugs serão descontados caso a caso.

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

Funcionalidade	Peso
Funcionalidade do TC3	2
Tiro	2
Bases inimigas	2
Mover inimigos	2
Finalização do Jogo	2

5 Erratas

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