

Enunciado do trabalho prático de CG 2010/11

No planeta XPTO existe um vale com 4 km quadrados onde se encontra um tesouro guardado num edifício localizado num extremo do vale. Este vale tem uma ondulação ligeira que pode atingir 30 metros no máximo.

O nosso agente encontra-se no outro extremo do vale e tem como objectivo alcançar o edifício para obter o tesouro. Para conseguir acesso ao edifício o nosso agente tem de recolher três chaves que estão dispersas pelo vale. Para localizar as chaves o nosso agente dispõe de um detector de metais com uma capacidade de detecção até 500 metros. O detector tem só um indicador que reporta a distância, mas não a direcção onde se encontra a chave.

A guardar o vale existem torres de protecção que detectam a presença de intrusos a uma distância de 1000 metros através de um sistema de radar, e conseguem apontar a sua arma à posição actual do intruso. A capacidade de fogo destas torres de protecção é de um tiro em cada 3 segundos. O tiro viaja a uma velocidade horizontal de 30 km/hora, e o agente desloca-se a uma velocidade de 10 km/hora. O tiro é disparada numa direcção paralela ao terreno, seguindo a sua ondulação sempre à mesma altura (1 metro).

Todas as quantidades acima descritos relativamente aos projecteis, número de torres, e velocidade do agente, devem ser configuráveis, sendo os valores apresentados utilizados por omissão.

Implemente uma aplicação em C/C++ utilizando OpenGL que reproduza o jogo acima descrito.

O projecto têm duas fases de implementação, com os seguintes requisitos:

Fase I (5 Valores)

- Na primeira fase o terreno pode ser plano;
- Implementar a navegação do agente no terreno utilizando uma câmara tipo First Person Shooter e uma câmara tipo Third Person Shooter (o jogador pode alternar entre as câmaras com o teclado) (1+1,5 valores);
- Dispor as chaves aleatoriamente e implementar o detector de chaves (0,5 valor);
- Carregar um modelo para o edifício do tesouro e para as torres de protecção. Orientar as torres de acordo com a posição do agente, desde que este esteja ao alcance da torre (1+1 valores);

Fase II (8 Valores)

- Implementar a criação do terreno a partir de um mapa de alturas (imagem em tons de cinza), e alterar a navegação do agente no terreno de forma a seguir a ondulação do terreno (2 valores);
- Realizar os disparos das torres, tendo em conta as especificações relativamente à capacidade de fogo e velocidade dos projecteis, e detectar quando um projectil disparado por uma torre atinge o agente (2 valores);

- Implementar View Frustum Culling para otimizar a aplicação relativamente ao edifício e torres (defina *axis aligned bounding boxes*, ou *bounding spheres* para estes itens) (2 valores).
- Utilizar Vertex Buffer Objects para o desenho do terreno (2 valores);

Na primeira fase deve ser entregue um relatório que inclua todo o código fonte.

Na segunda fase deve ser entregue um CD com duas pastas:

- uma com o código fonte e o projecto VS, Xcode ou makefile, e tudo o que for necessário para compilar a aplicação;
- uma segunda pasta com o executável e tudo o que for necessário para correr a aplicação, incluindo DLLs.

O CD deve ainda incluir o relatório final em PDF, devendo este também ser entregue em papel.

O relatório vale 4 valores da nota final. Os restantes 3 valores serão atribuídos com base na jogabilidade, qualidade da implementação, e visual.

Itens da primeira fase que não estejam implementados na altura da entrega da primeira fase não serão avaliados, mesmo que sejam implementados na segunda fase.

Datas de entrega:

Fase I –16 de Abril

Fase II –21 de Maio

As **apresentações** decorrem nas últimas semanas de aulas em calendário a afixar.

Os projectos devem ser realizados em grupos com o máximo de 5 alunos.