

Planejamento e Protótipo de Jogo 2D

**PUCRS - FACIN - Entretenimento
Digital**

03/2017

1 Enunciado

O trabalho **do semestre** consiste em planejar e desenvolver um *remake* de um jogo existente, usando a biblioteca *SFML*, ou *SFML+OpenGL*, em C++. O trabalho será dividido em duas etapas e as regras são as seguintes:

- O trabalho será desenvolvido em grupos de até **TRÊS** pessoas.
- O grupo deve implementar uma fase completa de um jogo já conhecido, presente na lista apresentada a seguir. Obviamente, não é necessário implementar absolutamente TUDO mas apenas um protótipo que contenha os elementos mais importantes do jogo.
- A implementação pode usar recursos de 3D (por exemplo, fornecidos por *OpenGL*), mas a jogabilidade deve ser **obrigatoriamente** em 2D.
- O jogo deve ser multiplataforma (*Linux/Windows/Mac*), ou seja, não deve usar recursos específicos de uma plataforma específica, apenas o que é fornecido pela *SFML* ou bibliotecas relacionadas.
- O jogo deve **obrigatoriamente** utilizar o framework

fornecido em aula. Se for utilizada alguma biblioteca além das mostradas em aula, o seu código deve ser obrigatoriamente multiplataforma também.

- O planejamento será realizado através de um *game design canvas* preenchido corretamente, um *mockup* de tela e um documento de design simplificado, que pode conter informações mais detalhadas que o *canvas*. Para um exemplo de design canvas preenchido e um *mockup*, [veja a apresentação neste link](#). Para editar um *canvas* vazio, faça uma cópia [desta apresentação](#).

O documento de design deve seguir o seguinte padrão (max. 2 páginas):

1. **Título do jogo**
2. **Introdução/história**
3. **Características (descrição, controles, *gameplay*)**
4. **Requisitos de arte e áudio:** *assets* são elementos gráficos ou sonoros utilizados em um jogo. Aqui deve-se fazer uma relação das imagens, sons e música que já foram ou que pretende-se utilizar, citando sua origem (criação própria, ou website, etc).
5. **Cronograma aproximado de desenvolvimento**

Observação: nem todos os elementos descritos na documentação precisam ser obrigatoriamente implementados na parte 1 do trabalho. Parte desses elementos (outros ambientes, parte do *gameplay*, ou parte da arte, etc), pode ser deixada para a parte 2.

2 Lista de Jogos

Escolha um dos jogos da lista, ou sugira alguma alternativa ao professor. Alternativamente, você pode experimentar transformar um jogo 3D em 2D, isto é, imaginar uma forma de converter a jogabilidade para 2D. Mas como isso exige um pouco mais de imaginação e coragem que o normal, converse antes com o professor se você resolver seguir por esse caminho.

- Kings Valley II (MSX) [\[Video\]](#)
- Mega Man (vários) [\[Video\]](#)
- River Raid (Atari 2600) [\[Video\]](#)
- Moon Patrol (arcade/outros) [\[Video\]](#)
- Donkey Kong (arcade/outros) [\[Video\]](#)
- Elevator Action (arcade) [\[Video\]](#)
- Ghost'n Goblins (arcade) [\[Video\]](#)
- Commando (arcade) [\[Video\]](#)
- Commander Keen (iD) [\[Video\]](#)
- Bubble Bobble (NES/outros) [\[Video\]](#)
- Lode Runner (vários) [\[Video\]](#)
- Frogger (Atari) [\[Video\]](#)
- Angry Birds (vários) [\[Video, versão Chuck Norris\]](#)
[\[Video, versão normal\]](#)

3 Entregas

- Cada grupo deverá **compactar os códigos-fontes, imagens e demais arquivos necessários**, com a implementação realizada em um arquivo **ZIP** com os nomes do grupo, **sem usar acentuação. Não**

envie RAR, 7Z, TAR.GZ ou qualquer outro formato!

- No mesmo arquivo .zip, deve ser incluído o *design canvas*, *mockup* de tela e *GDD* simplificado, em formato **PDF**.
- Deve ser feito o *upload* deste arquivo através do *Moodle* até às **17:35** do dia da entrega. Apenas um integrante do grupo deve fazer o *upload*.
- Todo o código entregue pelo grupo deverá conter um comentário no seu início, com o nome dos integrantes.

4 Avaliação

Os seguintes critérios de avaliação serão utilizados:

4.1 Parte 1 (15/05/17)

- A nota da parte 1 do trabalho será uma média ponderada entre a documentação entregue e a implementação parcial do trabalho em si:
$$T1 = 0,5 \cdot GDD + 0,5 \cdot Codigo$$
 - Ou seja, 50% da nota da documentação, mais 50% da nota da implementação.
- Para a composição da nota da implementação, serão avaliados os seguintes aspectos:
 - qualidade do código
 - funcionamento correto
 - *gameplay*
 - elementos de *design*/arte/áudio utilizados.

Observação: obviamente, como nosso background não é *design* nem arte, estamos apenas experimentando como uma equipe de verdade seria composta. Portanto, a avaliação do design, arte e áudio levará muito mais a criatividade e o empenho, do que a capacidade de *design* ou de arte demonstrada. Ou seja, é permitido utilizar arte (imagens, *sprites*, etc), áudio e outros elementos já prontos - desde que a sua origem seja citada.

4.2 Parte 2 (03/07/17)

- A nota da parte 2 do trabalho é dada pela implementação final do jogo:

$$T2 = \textit{Codigo}$$

- Ou seja, 100% da nota da implementação.

Observações Importantes:

- Os trabalhos serão apresentados ao professor no laboratório, na data e horário previstos. A ordem de apresentação seguirá a ordem de entrega no *Moodle*.
 - Os trabalhos que **não forem entregues** até o horário da aula, receberão **nota zero**.
 - Trabalhos que apresentarem **erro de compilação** receberão **nota zero**.
 - Trabalhos que apresentarem **cópia das soluções** de outros grupos, ou de exemplos encontrados na Internet, resultarão em **nota zero** para todos os alunos.
 - A presença no laboratório é **obrigatória**: os alunos que não estiverem presentes não receberão nota por este trabalho, mesmo que entreguem pelo *Moodle*.
-

Document generated by [eLyXer 1.2.5 \(2013-03-10\)](#) on
2016-08-15T14:30:46.070003