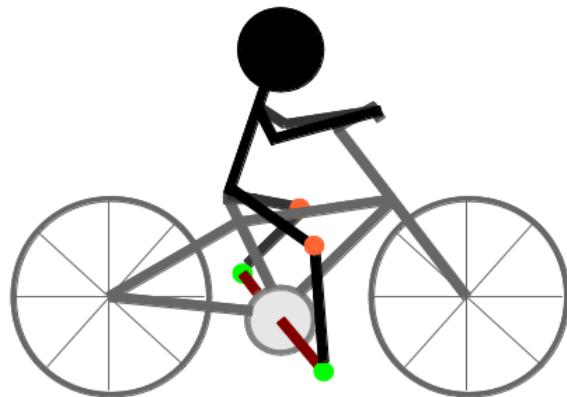


Universidade Federal de Santa Maria
Curso de Ciência da Computação
Disciplina: Computação Gráfica
Primeiro Semestre de 2019
Prof. Cesar Tadeu Pozzer Data: 30/04/2019

Trabalho 2 – Transformações geométricas



Ferramentas:

Linguagem C++, utilizando a API Canvas2D (disponível no [site da disciplina](#)) e IDE Code::Blocks, compilando com MinGW (disponível na [versão 17.12 da IDE Code::Blocks](#)).

Descrição:

Implemente um personagem em uma bicicleta com animação de pedalar, movendo (pelo menos) os pedais, as pernas e as rodas da bicicleta. As pernas devem se ajustar corretamente aos pedais conforme a animação ocorre. Você deve criar estruturas de que definam cada parte do personagem e matrizes de transformação para aplicar as animações ao personagem.

(não é necessário mover a bicicleta para frente, a não ser que você precise disso para algum bônus que implementar)

Requisitos do programa:

- Desenho da bicicleta e personagem (partes do desenho devem ser estruturadas em classes específicas).
- Animação de pedalar com posicionamento correto das pernas e pedais, e correta rotação das rodas.
- Uso de matrizes de rotação e translação para posicionar os elementos de desenho na tela.

Extras (para nota acima de 9,0):

- (+1) Usar **API Qt** e ferramenta **Qt Creator**, utilizando as funções de desenho da Canvas2D (demo gl_4_canvasQT).
- (+2) Ligação matematicamente correta entre o pedal e a perna.
- (+1) Utilizar curvas Bezier ou B-Spline para desenho do quadro da bicicleta.
- (+2) Implementar um chão sobre o qual a bicicleta anda, que deve interagir com as curvas no chão.
- (+1) Utilizar técnicas de parallax para desenhar um background com sensação de perspectiva conforme a bicicleta anda.
- (+1) Utilizar concatenação de matrizes de transformação.
- (+1) Mais ideias que acrescentem ao conteúdo explorado neste trabalho também poderão ser recompensadas.

Entrega:

Data de entrega:

14/05/2019, até as 23:59. Após esse prazo o trabalho será desconsiderado.

Formato de entrega:

No e-mail e no cabeçalho do arquivo, deve conter o nome completo do aluno. O arquivo deve ser enviado para pozzer3@gmail.com e adrian.k.47@hotmail.com, com o subject “CGT2”.

O programa deve ser enviado em um arquivo compactado fulano.RAR (fulano = login do aluno). Dentro deste arquivo deve haver um diretório com o mesmo nome do arquivo e, dentro deste diretório, os arquivos do trabalho. Deve-se enviar somente: código fonte (.cpp, .h, .hpp), imagens e arquivos de áudio (quando existirem) e o projeto (.cbp). Não devem ser enviadas libs, executáveis, DLLs, arquivos .pro.user, .suo e pastas de build.

Antes do envio, certifique-se de que o projeto contido no seu .rar funciona em qualquer diretório que ele seja colocado e não dependa de outros arquivos não incluídos no envio do trabalho, ou incluído nos demos do Pozzer.

Em caso de uso do QT, O diretório do projeto com os arquivos a serem entregues deverá ficar semelhante ao seguinte:

```

fulano/
    └── CanvasQT.pro
    ├── README.txt
    ├── glCanvas2d.cpp
    ├── glCanvas2d.h
    ├── main.c
    └── demais arquivo .cpp, .h e .hpp do seu projeto

```

Critérios de avaliação:

- Documentação: descrever no cabeçalho de cada arquivo a ideia geral do código e comentar o que cada método e classe faz.
- Clean code: estrutura do código e nomeação de métodos, classes e variáveis devem ser fáceis de ler e entender. Procurar manter o código o mais simples e organizado possível. Utilizar diferentes arquivos para diferentes classes.
- README: incluir um arquivo “README.txt” contendo informações sobre quais funcionalidades foram implementadas (requisitos e extras) e instruções de uso do programa caso o aluno julgue necessário ou caso tenha sido implementado uma funcionalidade extra que exija explicação.
- Pontualidade: Trabalhos não entregues na data não serão avaliados e receberão nota zero.
- Funcionalidade: o programa deve satisfazer todos os requisitos. Programas que não compilarem ou que não atenderem nenhum requisito receberão nota 0 (zero).

Você pode mandar e-mail para o monitor (adrian.k.47@hotmail.com) se tiver alguma dúvida relacionada a descrição ou implementação do trabalho.

Você pode discutir estratégias e ajudar o colega na implementação, porém evite passar código fonte. Programas semelhantes terão a nota 0 (zero).