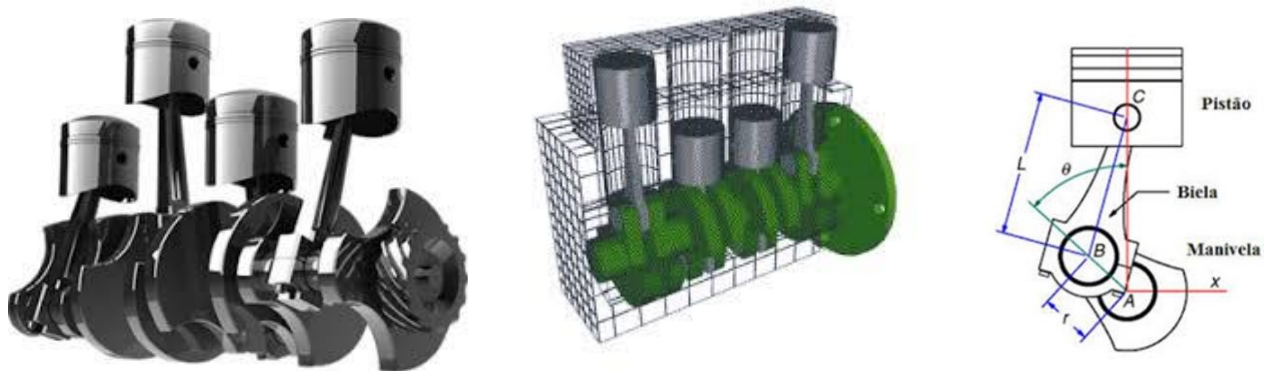


Trabalho 3 - Motor 3D a combustão Interna

Implemente um programa para fazer modelagem e visualização de um motor a combustão.

Linguagem C++, utilizando a API Qt (versão 5.10.0) e ferramenta Qt Creator, ou Canvas2D utilizando as funções de desenho da Canvas2D (demo gl_4_canvasQT) e compilando com MinGW (disponível na versão 17.12 da IDE Code::Blocks). Não podem ser utilizadas bibliotecas auxiliares. Não pode ser usada a API OpenGL.

O motor deve ter um virabrequim e pelo menos 1 pistão (com a biela e camisa).



O programa deve permitir a seleção do que vai ser exibido. Ex: virabrequim, camisa do pistão, pistão, biela, etc. Deve ser feito um encaixe preciso (matematicamente correto) da biela com o virabrequim e pistão. Veja os vídeos para entender o funcionamento. Não é necessário modelar o cabeçote.

https://www.youtube.com/watch?v=4W_NRHxekaY, https://www.youtube.com/watch?v=DKF5dKo_r_Y

Requisitos básicos (Max: 9.0)

- Rotação do eixo variável (RPM informada pelo usuário).
- Visualização ortográfica e perspectiva (sob vários ângulos) em wireframe.

Avançados

- Exibir vetores normais em cada face (até 1 ponto)
- Adição de mais pistões ao virabrequim
- Adição de detalhes aos componentes
- Motor em V
- Preenchimento de polígonos com iluminação por vértice (até 4 pontos)
- Preenchimento de polígonos com iluminação por pixel (até 5 pontos)
- Etc.

O trabalho deve apresentar uma lista de instruções, explicando de forma como o usuário deve interagir com o programa. Enumere no início do código fonte (arquivo main.cpp) os quesitos que foram implementados.

Data e Formato de Entrega:

- Data: 28/05/2018, até as 23:59. Após esse prazo o trabalho será desconsiderado.
- 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 bruno.t.nasc@gmail.com, com o subject “CG T3”.
- 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 (.pro). Não devem ser enviadas libs, executáveis, DLLs, arquivos .pro.user, .suo e pastas de build.
- Ex.: o arquivo fulano.rar deve conter um único diretório chamado fulano. Dentro desse diretório devem estar todos os arquivos do trabalho, incluindo o README.
- O diretório do projeto com os arquivos a serem entregues deverá ficar semelhante ao seguinte:

```
fulano/
├── CanvasQT.pro (caso for usado o QT)
├── 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 detalhes específicos de partes que mereçam uma explicação. Não comente por exemplo o que faz b++.
- README: incluir um arquivo “README.txt” contendo informações sobre quais funcionalidades foram implementadas (requisitos e extras).
- Pontualidade: Trabalhos não entregues na data não serão avaliados e receberão nota zero.
- Legibilidade: nome de variáveis, estruturação do código.
- Clareza: facilidade de compreensão – evite códigos complexos e desnecessários. Adote a solução mais simples possível.
- Funcionalidade: o programa deve satisfazer todos os requisitos. Programas que não compilarem ou que não atenderem nenhum requisito receberão nota 0 (zero).

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

Algumas coisas sobre a entrega desse trabalho para facilitar e evitar problemas na hora da correção.

- O projeto entregue deve estar em condições de ser compilado no Windows.
- O aluno deve implementar o seu trabalho sobre o demo `gl_4_canvasQT`. Esse demo já está com o projeto configurado e tem uma interface implementada na qual o aluno pode se basear para editar e fazer a interface do seu programa.
- Leiam atentamente toda a especificação do trabalho.