P3 CG

Gui Costa nº 61172

[Figura 1 - Gamma no geogebra 3](#_Toc68040328)

[Figura 2 - Vista inicial de 30 graus 4](file:///C:\Users\Gui\Documents\GitHub\CG\P3\README.docx#_Toc68040329)

[Figura 3 - Vista frontal quando se pressiona 'F' 4](file:///C:\Users\Gui\Documents\GitHub\CG\P3\README.docx#_Toc68040330)

[Figura 4 - Vista traseira quando se pressiona 'A' 4](file:///C:\Users\Gui\Documents\GitHub\CG\P3\README.docx#_Toc68040331)

[Figura 5 - Vista de cima quando se pressiona 'C' 4](file:///C:\Users\Gui\Documents\GitHub\CG\P3\README.docx#_Toc68040332)

[Figura 6 - Vista de baixo quando se pressiona 'B' 5](file:///C:\Users\Gui\Documents\GitHub\CG\P3\README.docx#_Toc68040333)

[Figura 7 - Vista lateral esquerda quando se pressiona 'E' 5](file:///C:\Users\Gui\Documents\GitHub\CG\P3\README.docx#_Toc68040334)

[Figura 8 - Vista lateral direita quando se pressiona 'D' 5](#_Toc68040335)

Objetivos:

* mostrar o símbolo colorido do trabalho P2 em 3D;
* Adicionar profundidade, sendo que os vértices devem considerar um espaço 3D;
* visão ortogonal em 6 vistas mudando de vista com pressão de teclas (E - lateral esquerda, D - lateral direita, F - frente, A - atrás, C - cima, B - baixo) o símbolo deve ser colocado na cena com um ângulo de visão de 30 graus;

As cores foram modificadas relativamente a última versão para melhor mostrar a profundidade da letra.

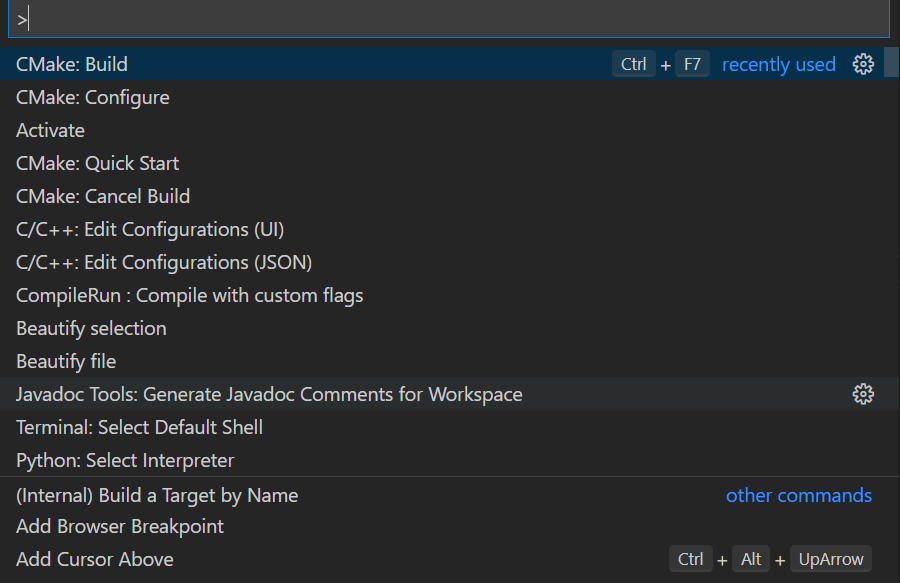
# Plataforma de desenvolvimento:

SO Windows com MINGW64, editor de texto é o visual code com extensão CMAKE\_TOOLS para a execução dos ficheiros CMakelist.txt. A linguagem usada é c++. As livrarias usadas são GLAD e GLFW 3.3.3 com adição de GLM para operações matriciais. Github para a gestão de versões. A ferramenta usada para modelar foi geogebra.

# Resultados:

Build:

Devido ao uso do vscode e das extensões relativas ao cmake o processo é bastante simples, para cada pasta nova (neste caso o P3) abrimos a palete de comandos do vscode visíveis na imagem a baixo e corremos primeiro CMake: configure que executa o seguinte comando

"C:\Program Files\CMake\bin\cmake.EXE" --no-warn-unused-cli -DCMAKE\_EXPORT\_COMPILE\_COMMANDS:BOOL=TRUE -DCMAKE\_BUILD\_TYPE:STRING=Debug "-DCMAKE\_C\_COMPILER:FILEPATH=C:\Program Files\mingw-w64\x86\_64-8.1.0-posix-seh-rt\_v6-rev0\mingw64\bin\gcc.exe" "-DCMAKE\_CXX\_COMPILER:FILEPATH=C:\Program Files\mingw-w64\x86\_64-8.1.0-posix-seh-rt\_v6-rev0\mingw64\bin\g++.exe" -Hc:/Users/UserName/Documents/GitHub/CG -Bc:/Users/Gui/Documents/GitHub/CG/build -G "MinGW Makefiles"

Depois de realizar o projeto e querer testar corre-se o segundo comando da palete que é o CMake: Build que vai compilar o projeto e as dependências, este executa a seguinte linha

"C:\Program Files\CMake\bin\cmake.EXE" --build c:/Users/UserName/Documents/GitHub/CG/build --config Debug --target P3 -- -j 18

Run:

Finalmente para executar escreve-se "C:\Users\UserName\Documents\GitHub\CG\build\P3\P3.exe"

O programa executa com sucesso e gera uma janela de fundo cinzento-escuro que mostra a letra gamma maiúscula:

Imagens:

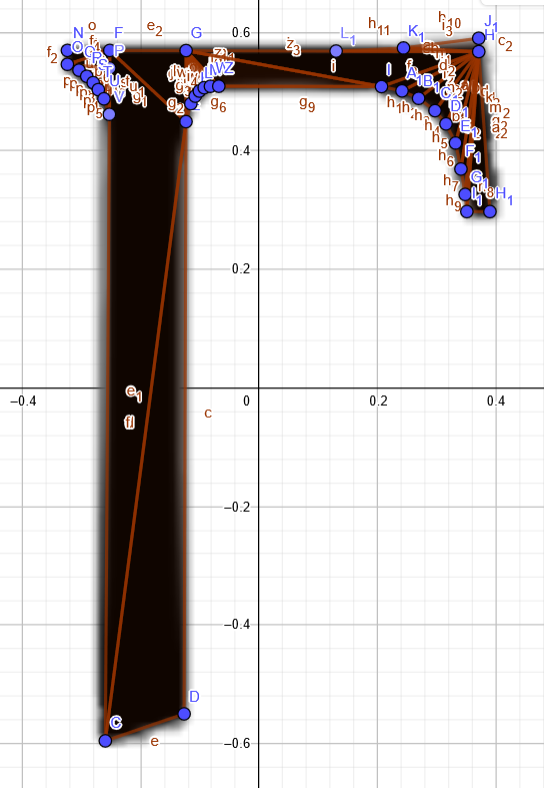


Figura 1 - Gamma no geogebra

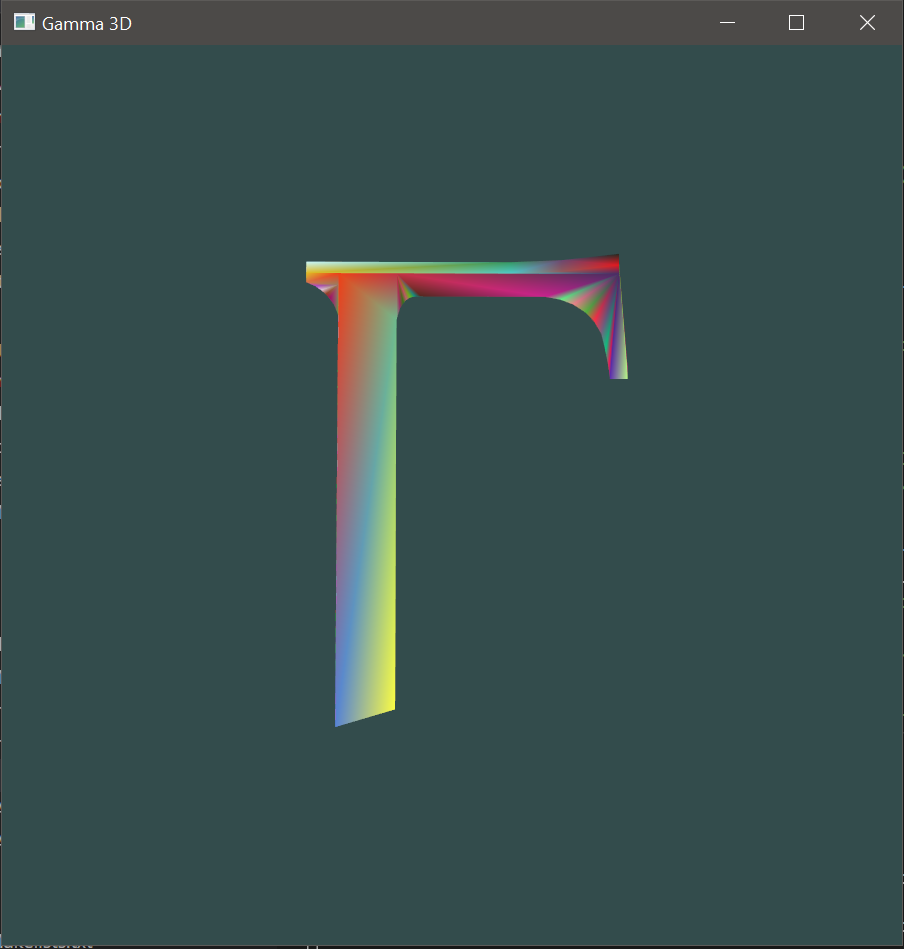
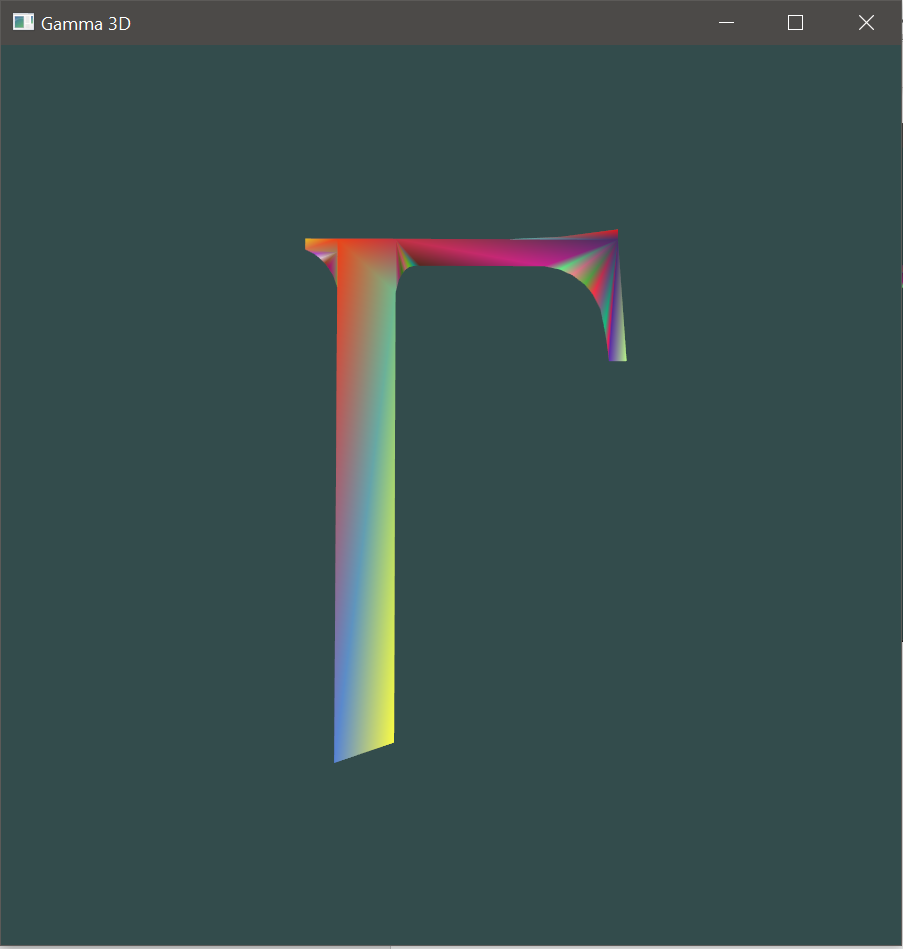


Figura 2 - Vista inicial de 30 graus

Figura 3 - Vista frontal quando se pressiona 'F'

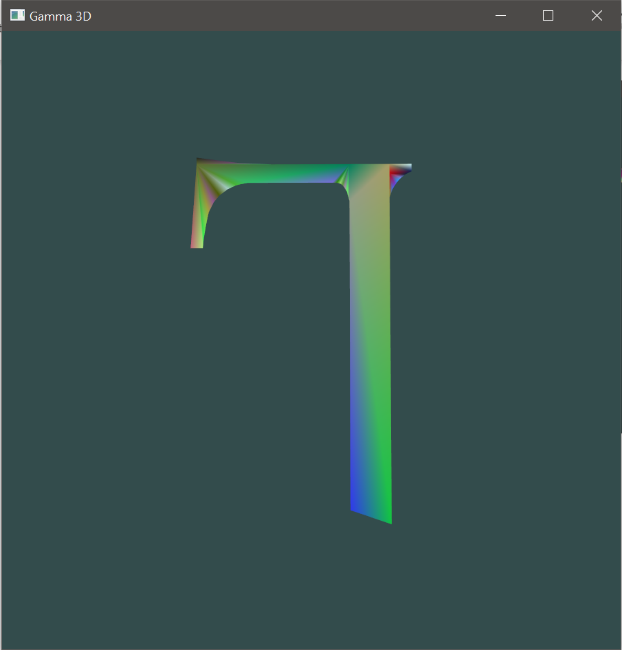
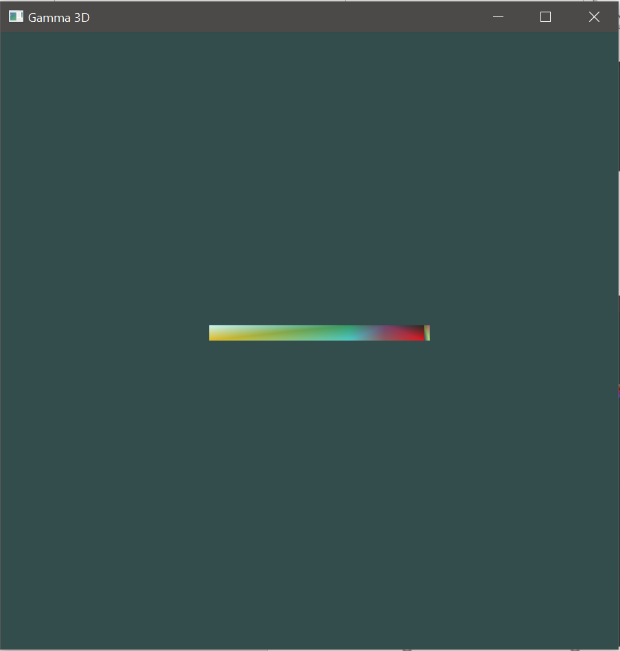


Figura 4 - Vista traseira quando se pressiona 'A'

Figura 5 - Vista de cima quando se pressiona 'C'

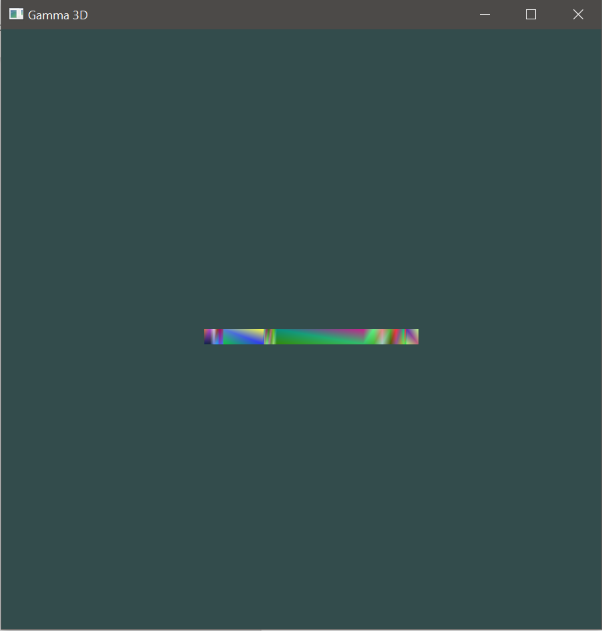
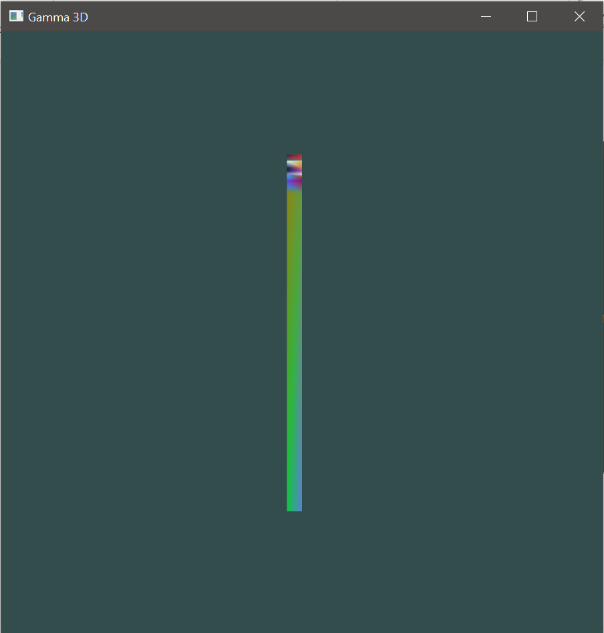


Figura 6 - Vista de baixo quando se pressiona 'B'

Figura 6 - Vista de baixo quando se pressiona 'B'

Figura 7 - Vista lateral esquerda quando se pressiona 'E'

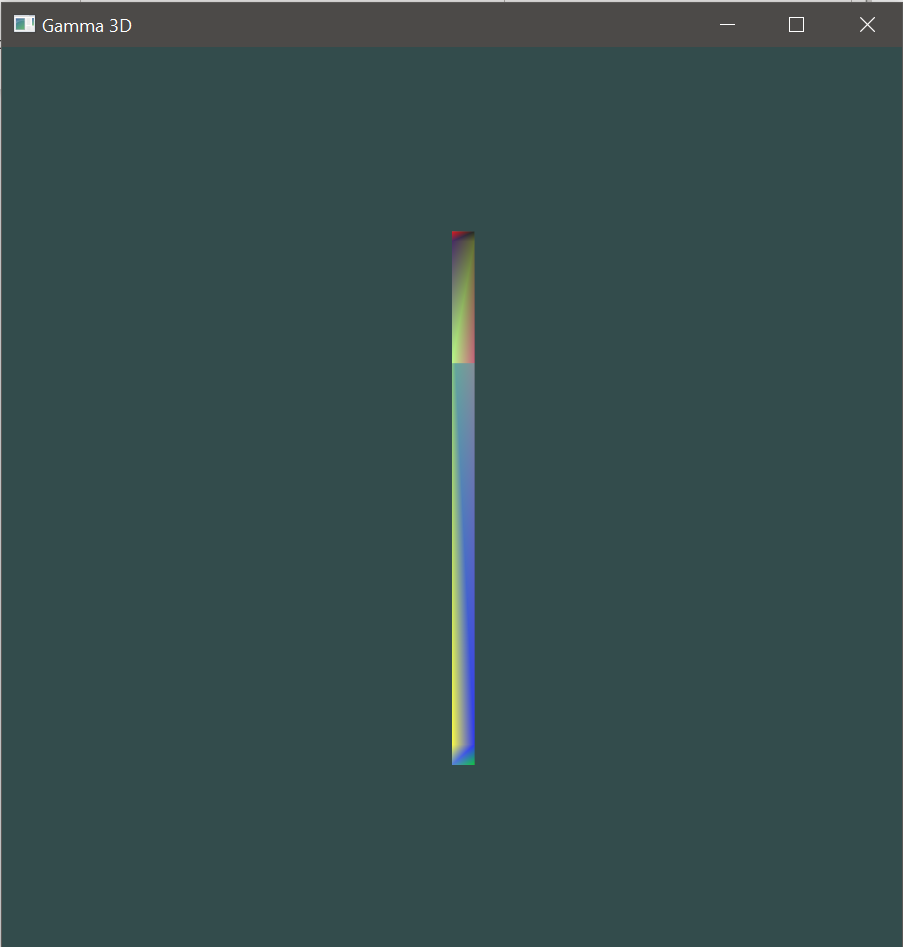


Figura 8 - Vista lateral direita quando se pressiona 'D'