

”

**E-fólio B** | Folha de resolução para E-fólio

**UNIDADE CURRICULAR:** Computação Gráfica

**CÓDIGO:** 21020

**DOCENTE:** António Araújo

**A preencher pelo estudante**

**NOME:** Hernâni Filipe Resendes Coelho

**N.º DE ESTUDANTE:** 1800045

**CURSO:** Engenharia Informática

**DATA DE ENTREGA:** 28/12/2021

## TRABALHO / RESOLUÇÃO:

Este trabalho tem como objetivo apresentar uma página web onde é possível visualizar graficamente a aplicação do algoritmo da curva bezier cúbica para desenhar uma curva bezier aplicada a segmentos tubulares, mediante os pontos de controlo que transitam como argumento da função. Para o seu desenvolvimento foi utilizado as linguagens de programação HTML, CSS e JavaScript, e as bibliotecas three e orbitcontrols.

A organização da estrutura de ficheiros da página está organizada de acordo com as indicações, pelo que resulta da seguinte forma:

- Diretório base
  - Index.html
  - bezier.mjs
  - readMe.pdf
  - src
    - js
      - init.mjs
      - main.mjs
    - css
      - style.css

### Organização do código:

O código está organizado de forma modular, de forma a otimizar a legibilidade, evitando que referências a linguagens diferentes constem do mesmo ficheiro, assim todas estas referências são invocadas de forma externa. Por uma questão de otimização de esforços, foi reutilizado uma parte do código do efólio A. Alterando cirurgicamente alguns pontos já existentes e adicionando novas features que permitem concluir com o objetivo.

O Ficheiro index.html consiste num ficheiro simples que tem apenas a finalidade de fazer a importação do módulo inicial “init.mjs”, bem como adicionar formatação específica, css, aos elementos de informação textual da página web. Assim que a janela do browser abrir, este módulo invoca a função “init” para iniciar a aplicação da página. Todas as funções da aplicação estão definidas no módulo “main.mjs”.

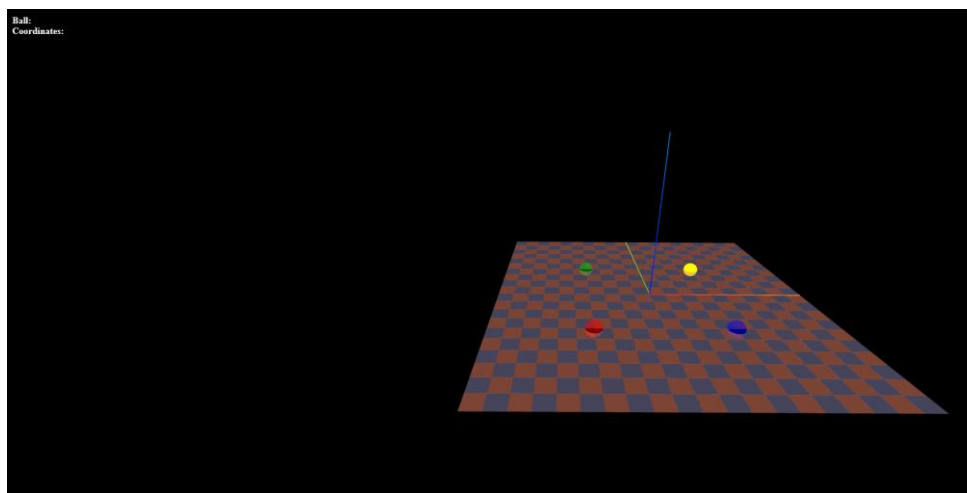
A função “init” chama as funções para criar a cena, criar o plano de 20 por 20 quadrados com cor alternada, criar as linhas do referencial tridimensional, adicionar as bolas referentes aos pontos c0, c1, c2 e c3 e posicionar a camara na cena. O utilizador tem a liberdade de utilizar o rato para se deslocar no plano.

A aplicação recorre ao uso de handlers de eventos para tratar determinados eventos que são despoletados. Assim, o evento do movimento do rato é detetado e tratado pela função “onMouseMove”, que deteta e regista as coordenadas normalizadas. Dependendo da finalidade, estas coordenadas (float), podem apenas informar para a consola ou então serem guardadas como as coordenadas dos pontos de controlo da curva bezier. Existe também outro handler para tratar dos eventos de teclas pressionadas, e assim que uma tecla é pressionada é invocado a função “keyboardPress” que posteriormente, conforme a tecla, invoca a função correspondente às ações definidas no enunciado. caso seja a tecla x, é invocada a função “bezierCurveDraw” para desenhar a bezier cúbica dos pontos de controlo definidos. Inicialmente é definido um objeto pela classe “bezierCurve” que herda da classe base intrínseca da biblioteca THREE, Curve. A classe definida possui um método que foi adaptado “getPoint” para invocar a função “bezier3” e assim calcular as coordenadas da curva para cada instante de t,  $0 \leq t \leq 1$ . Posteriormente, retorna um objeto com todas as coordenadas dos pontos para cada instante de t, após a posse de todos os pontos é possível construir a bezier, sendo esta um objeto da classe “tubeGeometry”.

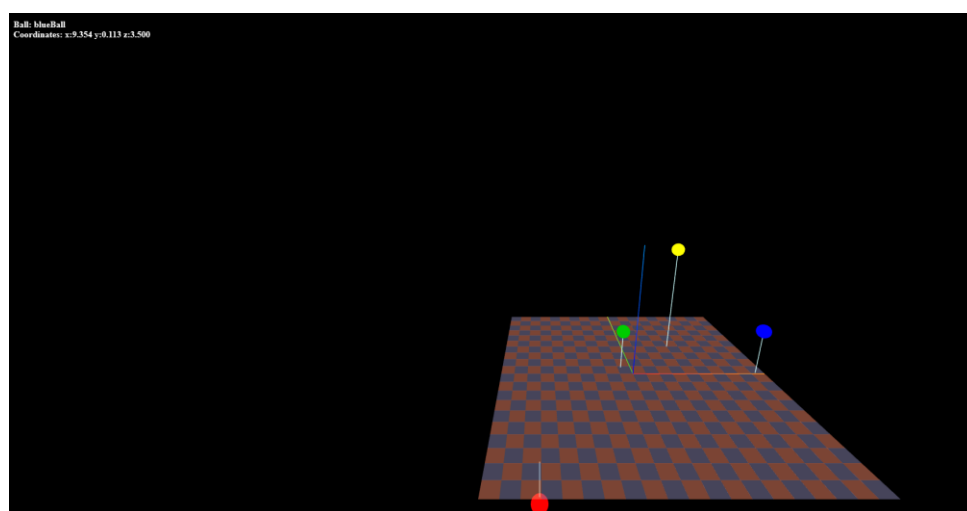
A função “bezier3” aplica a expressão do caso geral de uma bezier curve, para cada coordenada, x, y e z.

$$p(t) = c_0(1-t)^3 + c_1.3t(1-t)^2 + c_2.3t^2(1-t) + c_3.t^3$$

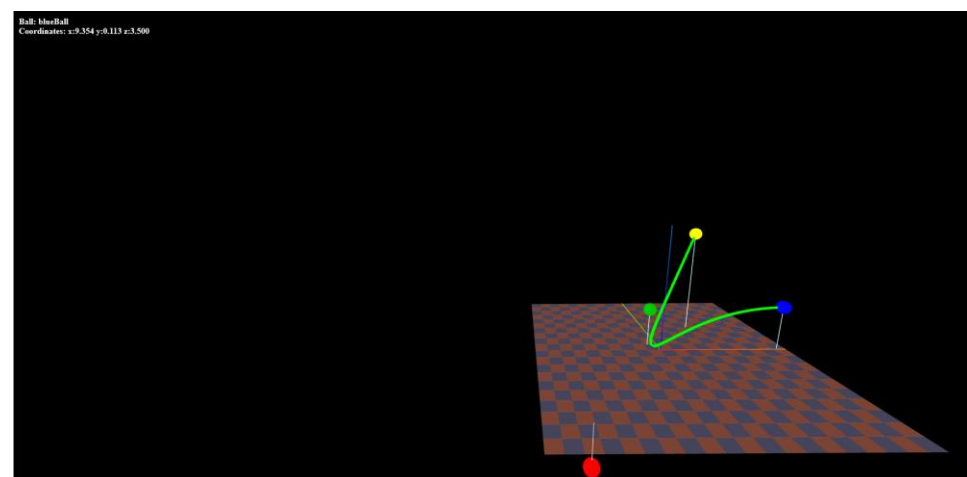
Testes executados com o live Server do Visual Studio na porta 5500.



*Figura 1 Posição inicial da aplicação com foco no ponto (0,0)*



*Figura 2 Definição das coordenadas dos pontos c0, c1, c2 e c3*



*Figura 3 Execução da bezier curve, para os pontos definidos anteriormente*

Bibliografia:  
Documentação disponibilizada no âmbito da disciplina