

# Computação Gráfica (MIEIC)

## Trabalho Prático 6


### *Projeto Final*



## Objetivos

- Aplicar os conhecimentos e técnicas adquiridas até à data
- Utilizar elementos de interação com a cena, através do teclado e de elementos da interface gráfica

## Trabalho prático

Ao longo dos pontos seguintes são descritas várias tarefas a realizar. Algumas delas estão anotadas

com o ícone  (captura de imagem). Nestes pontos deverão, com o programa em execução, capturar uma imagem da execução. Devem nomear as imagens capturadas seguindo o formato "CGFImage-tp6-TtGgg-x.y.png", em que TtGgg referem-se à turma e número de grupo e x e y correspondem ao ponto e subponto correspondentes à tarefa (p.ex. "CGFImage-tp6-T3G10-2.4.png", ou "CGFImage-tp6-T2G08-extra.jpg").

Nas tarefas assinaladas com o ícone  (código), devem criar um ficheiro .zip do vosso projeto, e nomeá-lo como "CGFCode-tp6-TtGgg-x.y.zip", (com TtGgg, x e y identificando a turma, grupo e a tarefa tal como descrito acima). Quando o ícone  surgir, é esperado que executem o programa e observem os resultados. No final, devem submeter todos os ficheiros via Moodle, através do link disponibilizado para o efeito. Devem incluir também um ficheiro **ident.txt** com a lista de elementos do grupo (nome e número). Só um elemento do grupo deverá submeter o trabalho.

## Preparação do Ambiente de Trabalho

Este trabalho deve ser baseado nos vários elementos básicos criados nas aulas anteriores (**LightingScene.js**, planos, cubos, semi-esferas e cilindros texturados).

Iremos acrescentar uma classe de interface que criará uma área de interface gráfica com alguns elementos de interação, e que será também responsável por gerir eventos de teclado. Para tal, é fornecido o ficheiro **MyInterface.js** que devem incluir no projeto da seguinte forma:

- Colocar o ficheiro na mesma diretoria dos restantes ficheiros Javascript do projeto
- editar o ficheiro **main.js** e
  - adicionar '**MyInterface.js**' à lista de ficheiros a incluir
  - substituir no código da função main a referência a **CGFInterface** por **MyInterface**
- Editar o vosso ficheiro de cena (**LightingScene.js**) e
  - acrescentar no método **LightingScene.init**, as seguintes variáveis:

```
this.option1=true; this.option2=false; this.speed=3;
```
  - acrescentar ao ficheiro o seguinte método:

```
LightingScene.prototype.doSomething = function ()  
{ console.log("Doing something..."); };
```

# 1. Criação da classe **MySubmarine** e ambiente inicial

Neste exercício procura-se criar uma geometria para representar um submarino que servirá de *avatar*.

1. Crie uma classe **MySubmarine** que represente um submarino. Esta classe será responsável, inicialmente, pelo desenho do submarino. Nesta fase inicial do trabalho será constituída apenas por um triângulo paralelo ao plano XZ, com coordenadas (0.5, 0.3, 0), (-0.5, 0.3, 0), (0, 0.3, 2), ou seja, um triângulo a apontar para +ZZ.
2. Altere a cor de fundo da cena para azul.
3. Crie um plano e aplique-lhe uma textura que simule o fundo oceânico; optimize a utilização da textura, fazendo-a repetir várias vezes ao longo do plano.
4. Coloque o relógio num poste na posição ( 8, 5, 0).
5. Aplique as transformações necessárias para colocar o submarino no centro da cena, a apontar (aproximadamente) para o poste.

(1.5  ) (1.5  ) 

## 2. Controlo do Submarino

Neste exercício procura-se criar um mecanismo de controlo para o avatar acima criado. Consulte a classe **MyInterface.js** para ver exemplos de utilização que ajudarão na resolução destes pontos. Estudar a biblioteca javascript dat.GUI através dos seguintes exemplos: <https://workshop.chromeexperiments.com/examples/gui/#1--Basic-Usage>

1. Crie um mecanismo para controlar o submarino utilizando as teclas: rodar para a esquerda ou para a direita, conforme a tecla premida é "A" ou "D", e mover-se no sentido para onde está virado ou no sentido contrário, conforme se pressione "W" ou "S", respectivamente. Para este efeito, deve criar as variáveis ou métodos na classe da cena de forma a poder alterá-las ou invocá-las na classe de interface.

(2.1  ) 

## 3. GUI

Neste exercício procura-se criar uma interface gráfica (GUI) com alguns controlos para alterar parâmetros da cena em tempo de execução.

1. Adicione à GUI um grupo intitulado "Luzes" (remova/comente/substitua o grupo do exemplo). Acrescente ao novo grupo, por cada fonte de luz utilizada, uma checkbox. Cada checkbox (estado on/off) deve permitir alterar o estado (respetivamente acesa/apagada) da fonte de luz que lhe diz respeito.



2. Adicione um botão que pause/retome o mecanismo de animação do relógio da cena;

(3.2  ) (3.2  ) 

## 4. Modelação do Submarino

Neste exercício deverá criar o modelo do submarino com mais detalhe, para substituir a geometria usada no ex. 1. O submarino será composto por vários elementos, tal como descrito abaixo e representado na figura 1 (dimensões aproximadas). Os elementos devem poder ter texturas aplicadas, texturas essas que deverão ser criadas/selecionadas pelos estudantes.

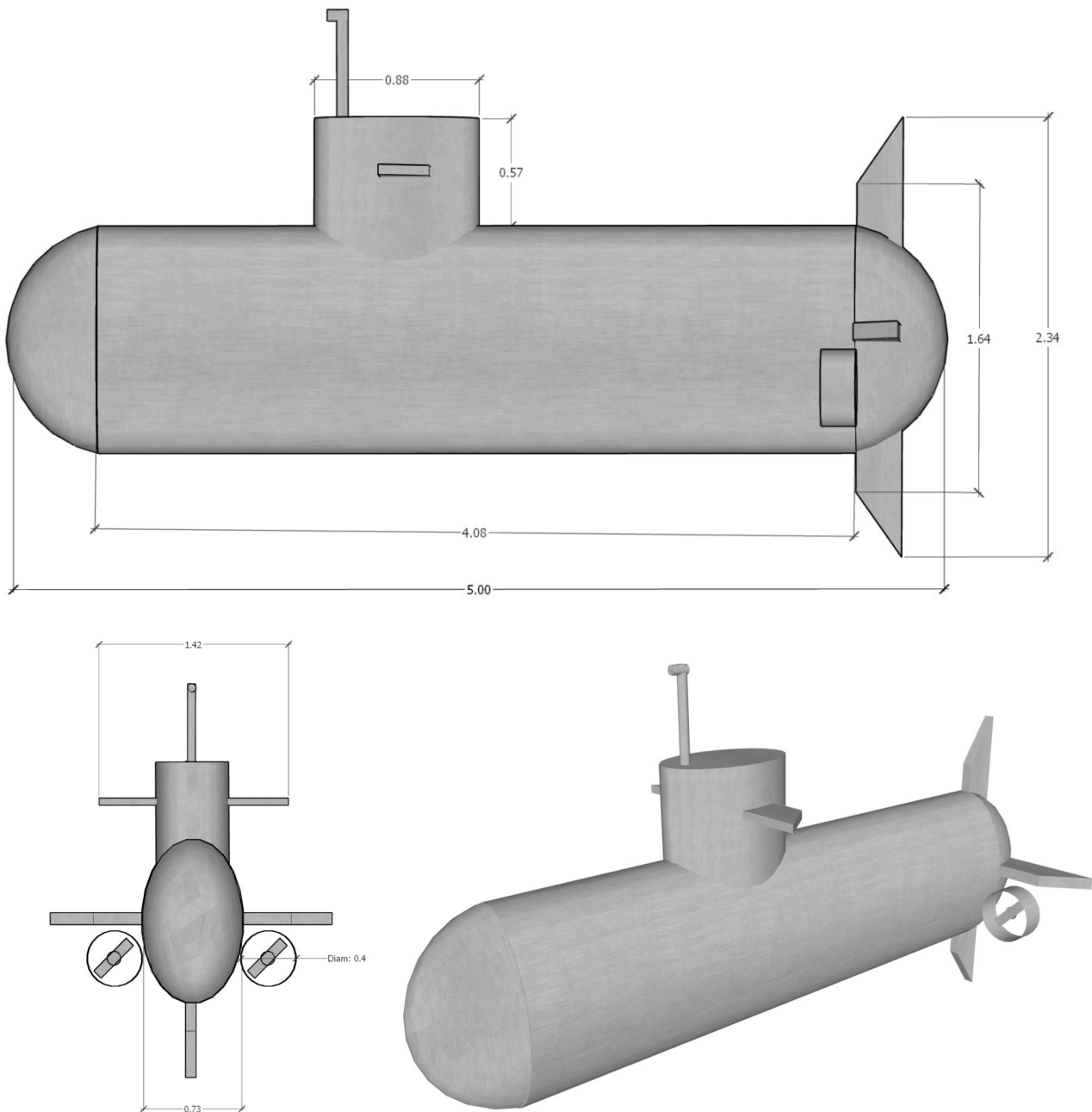


Figura 1: Vista de lado, de trás e em perspetiva do submarino, com dimensões indicativas aproximadas

1. Construa o submarino com os seguintes componentes:
  - um cilindro e duas (semi-) esferas para o corpo principal, e um cilindro com topo para a torre superior, tudo "comprimido" no eixo dos xx;
  - dois cilindros para o periscópio;
  - três trapézios para as "barbatanas" da torre e da parte traseira (cada um atravessa o corpo de um lado ao outro);

- para as hélices, um paralelepípedo com uma semi-esfera no centro; à volta, um cilindro com faces interiores e exteriores, sem topos.

(4.1  ) 

2. Construa uma interface para a seleção das texturas, integrada na GUI da aplicação. Deve para o efeito usar um controlo do tipo “drop-down”. Para implementar este tipo de controlos, sugere-se:

- Declarar na cena um array ***submarineAppearances*** que contenha as várias appearances possíveis
- Declarar um dicionário ***submarineAppearanceList*** que mapeie as strings identificando cada appearance ao seu índice em ***submarineAppearances***.
- Declarar na cena uma variável ***currSubmarineAppearance*** que identifique o índice da appearance selecionada/atual
- Adicionar um controlo na interface que fique associado a ***currSubmarineAppearance*** e a ***submarineAppearanceList***  
(exemplos em <http://workshop.chromeexperiments.com/examples/gui/#2--Constraining-Input>)
- Ajustar o código de desenho da cena ou do robot para que seja usada a *appearance* correta.

3. Execute e altere as texturas usando a GUI

(4.3  ) (4.3  ) 

(continuação brevemente...)