

# Programação

## Primeiro trabalho prático Semestre de Inverno de 2021/2022

Cada grupo terá que entregar até 15 de novembro, no respetivo *site* da turma, os ficheiros (extensão .kt) com o código do trabalho devidamente indentado e comentado.

O trabalho consiste em fazer um programa em *Kotlin* para desenhar a cabeça da mascote do [Android](#) tal como mostra a figura 1. O desenho pode ser redimensionado usando as teclas (+ e -) e pode ser deslocado com as teclas (←, →, ↓ e ↑) ou com o *click* do rato.

A cabeça do android é desenhada com as a proporções indicadas na figura 2, em que todos os elementos são calculados em função de **R** (raio do semicírculo da cabeça). Por exemplo, a espessura da linha que desenha cada antena é de **0,08 R**, que no caso de **R=60** é **4,8**, correspondendo a **5 pixels** depois de convertido com arredondamento para um valor do tipo `Int` com a função `roundToInt()`.

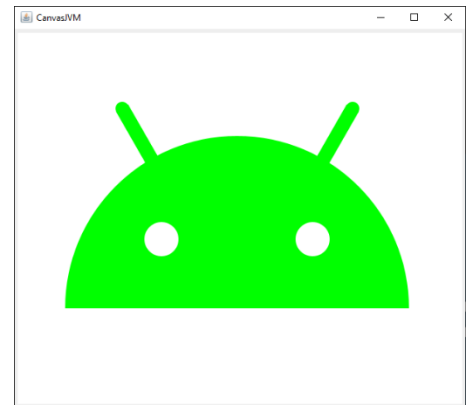


Figura 1 : Janela CanvasJVM

Inicialmente, o centro do semicírculo da cabeça fica centrado no *canvas* e o raio (**R**) tem 100 pixels.

A cabeça deve ser representada por um valor do tipo agregado `Head` definido como:  
`data class Head(val x :Int, val y :Int, val radius :Int)`  
em que **x** e **y** é centro do semicírculo e **radius** é o raio do semicírculo.

O programa só pode ter um ponto de mutabilidade, ou seja, só deve ter uma variável (**var**) do tipo `Head` cujo valor é alterado por cada modificação da representação da figura.

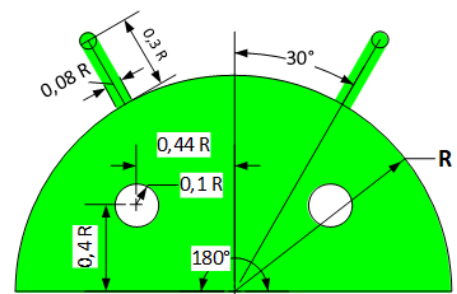


Figura 2: Proporções do desenho

Quando é premida a tecla (+) o raio é aumentado 1 pixel e a figura é redesenhada com as novas proporções. Quando é premida a tecla (-) o raio é reduzido 1 pixel e a figura também é redesenhado com as novas proporções, mas só no caso da espessura da linha das antenas ficar com um valor maior ou igual a 1 pixel.

Quando for premida uma das teclas de cursor (←, →, ↓ ou ↑) o valor de **x** ou de **y** do centro do semicírculo é aumentado ou reduzido em 4 pixels. Quando for dado um *click* com rato na área do *canvas* o centro do semicírculo é alterado para a posição do *click*. Em ambos os casos, a figura é redesenhada no novo centro.

Na realização do trabalho devem ser respeitadas as seguintes regras:

- Evitar mutabilidade. Este programa só necessita de um ponto de mutabilidade, já referido anteriormente.
- Não repetir código. Definir funções que são chamadas em diversos locais.
- Não fazer funções demasiado extensas. Decompor funções extensas em várias funções. Para ser compreensível, cada função dever ter menos do que 25 linhas, aproximadamente.
- Não repetir valores com o mesmo significado nem usar “valores mágicos”, em que não se percebe porque estão ali. Uma determinada característica só deve estar expressa num local e esse deve ser o local esperado. Caso seja necessário, definir valores nomeados, cujo nome é esclarecedor e será usado num ou vários locais do programa. Assim, quando for necessário alterar esse valor, só será necessário modificar essa definição.

Por exemplo, definir o valor de deslocamento provocado pelas teclas de cursor com a declaração global:

```
const val DELTA_CURSOR = 4
```

Uma implementação do programa pretendido está disponível no ficheiro [trab1.jar](#). Para executar este programa, provavelmente basta mandar executá-lo no seu sistema operativo, mas, em alternativa, abra uma janela de comandos localizada onde reside o ficheiro e execute o comando: `kotlin trab1.jar` ou então o comando: `java -jar trab1.jar`

Bom trabalho.