

## Relatório do trabalho de Linguagens de Programação



## Atendedor de Chamadas

*José Miguel Silva*

[180100265@esq.ipsantarem.pt](mailto:180100265@esq.ipsantarem.pt)

## Resumo

O **atendedor** de chamadas é uma aplicação que tem como objetivo fornecer ajuda ao Covid-19 com utilização de **árvore binária** em Java.

## Introdução

Com a realização deste trabalho pretendi desenvolver um atendedor de chamadas de ajuda ao Covid-19.

O utilizador (a pessoa que faz a chamada) vai seleccionando as opções determinadas pela sua condição de saúde e vai percorrendo a árvore entre a direita e a esquerda.

As chamadas são guardadas e no final temos um histórico das mesmas.

Consoante os sintomas que o utilizador tenha, no final, pode ser redirecionado para uma equipa de apoio ao Covid-19.

Diagrama

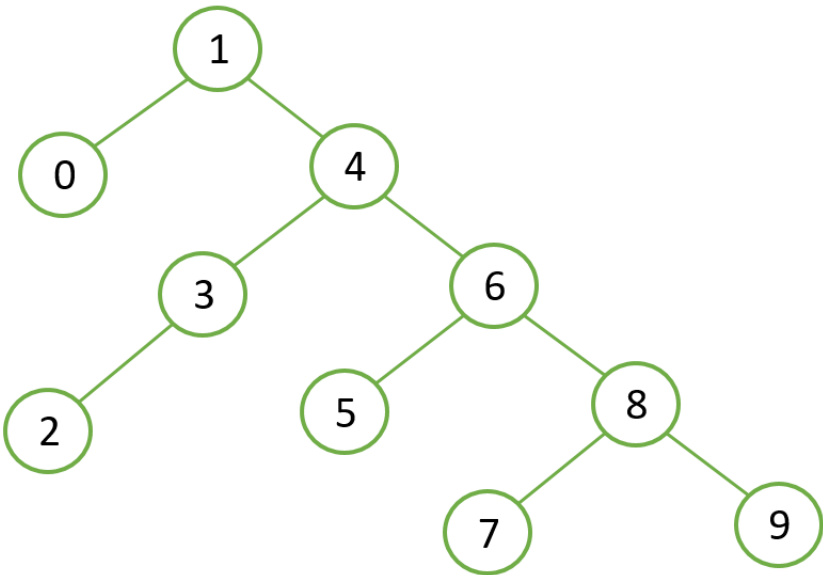


Figura 1 - Árvore Binária

# Linguagem

A linguagem de programação utilizada neste projeto foi Java



## Java

A função “imprimirResumo” recebe um *array* com o resumo da chamada e no fim apresenta-nos o resumo da mesma.

```
public void imprimirResumo(String [][] resumoChamada) {  
  
    System.out.println("");  
    System.out.println("-----");  
    System.out.println("Resumo da Chamada:");  
  
    for(int i = 0; i < resumoChamada.length; i++) {  
  
        if(resumoChamada[i][1] == "sim")  
            System.out.println(resumoChamada[i][0]);  
    };  
    System.out.println("Chamada terminada.");  
    System.out.println("-----");  
}
```

A função “mensagemAuto” recebe um *array* para guardar o resumo da chamada e uma variável do tipo *Boolean*.

Esta função, consoante o nó da árvore em que nos encontramos, dar-nos-á a mensagem correspondente.

(tirei print ao cabeçalho porque a função é grande)

```
public String mensagemAuto(String [][] resumoChamada, boolean fim) {  
    int valor=raiz.getValor();  
    String mensagem = "";  
  
    switch(valor) {
```

A função “atendedorHandler” recebe o valor premido pelo utilizador que faz a chamada, esta função está incluída na classe árvore e por isso conhece a sua raiz. Chama então a função “atendedorHandler” mas desta vez a privada e verifica se o valor recebido pela função está à direita ou esquerda da raiz em que nos encontramos.

```
public boolean atendedorHandler(int valor) {  
    return atendedorHandler(raiz, valor);  
}  
  
private boolean atendedorHandler(NoBinTree no, int valor) {  
    if(no.getFilhoEsquerda() != null && no.getFilhoDireita() != null)  
        if(no.getFilhoEsquerda().getValor() == valor || no.getFilhoDireita().getValor() == valor)  
            return true;  
  
    return false;  
}
```

A função “mudarRaiz” recebe o valor premido pelo utilizador que faz a chamada e tem como função percorrer a árvore consoante o valor recebido.

```
public void mudarRaiz(int valor) {  
    if(this.raiz.getFilhoEsquerda().getValor() == valor)  
        this.raiz = this.raiz.getFilhoEsquerda();  
    else  
        this.raiz = this.raiz.getFilhoDireita();  
}
```

## O resultado

Foi criado um atendedor de chamadas que guia os utilizadores em relação ao Covid-19. Informa de como ficar seguro do Covid-19, bem como faz uma pequena triagem dos sintomas que o utilizador tem e faz um possível encaminhamento.

## Conclusão

O trabalho vai de encontro ao que foi lecionado e adquirido a nível de aprendizagem na UC de Linguagens de Programação.

## Bibliografia

Moodle IPSantarém – Linguagens de Programação

<https://moodle.esgt.ipsantarem.pt/course/view.php?id=1512>