

# Linguagem de Programação II

## Trabalho Prático

### Grupo

---

a18832	Ricardo Silva
a15255	Pedro Macedo



Licenciatura em Engenharia de Sistemas Informáticos  
Instituto Politécnico do Cávado e do Ave

## Conteúdo

<b>1</b>	<b>Introdução</b>	<b>3</b>
<b>2</b>	<b>Problema</b>	<b>3</b>
<b>3</b>	<b>Solução</b>	<b>3</b>
<b>4</b>	<b>Utilização</b>	<b>4</b>
<b>5</b>	<b>Arquitetura</b>	<b>4</b>
5.1	Classes . . . . .	5
5.2	Estrutura das Classes . . . . .	6
<b>6</b>	<b>Exemplo de Execução</b>	<b>7</b>
<b>7</b>	<b>Problemas encontrados</b>	<b>8</b>
<b>8</b>	<b>Conclusão</b>	<b>8</b>

## 1 Introdução

O objetivo deste trabalho prático consiste em melhorar os nossos conhecimentos já iniciados em LP1 mas numa nova linguagem multiparadigma **C#**. Todo o trabalho desenvolvido neste trabalho prático irá ser melhorado e alterado com o decorrer das aulas que visam em fazer com que os trabalhos sejam melhorados gradualmente.

## 2 Problema

O problema neste trabalho prende-se na criação de um sistema de gestão de um hospital, onde se gere a entrada e saída de paciente anexando todos os dados sobre os mesmos.

## 3 Solução

A solução passa pela criação de várias funções auxiliares de modo a organizar todos esses dados e utilizá-los para criar a ficha do paciente.

## 4 Utilização

De modo a correr o programa é apenas necessário compilar e executar o ficheiro `.exe` nesta primeira fase este trabalho funciona com dados teste, ou seja dados já introduzidos de maneira a verificar se o programa corre como planeado.

## 5 Arquitetura

A aplicação desenvolvida está dividida em diferentes *funções* e várias classes como podemos ver na Figura 1, as quatro classes presentes neste trabalho : doentes, hospital, infeção e pessoa.

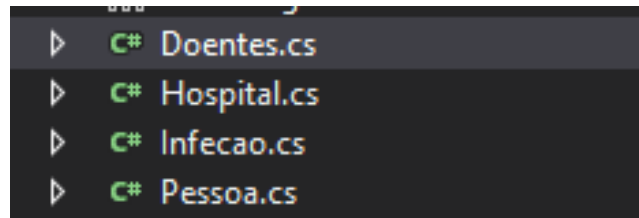


Figura 1: Classes.

## 5.1 Classes

Essas quatro classes estão ligadas, formando um diagrama de classes representado na Figura 2.

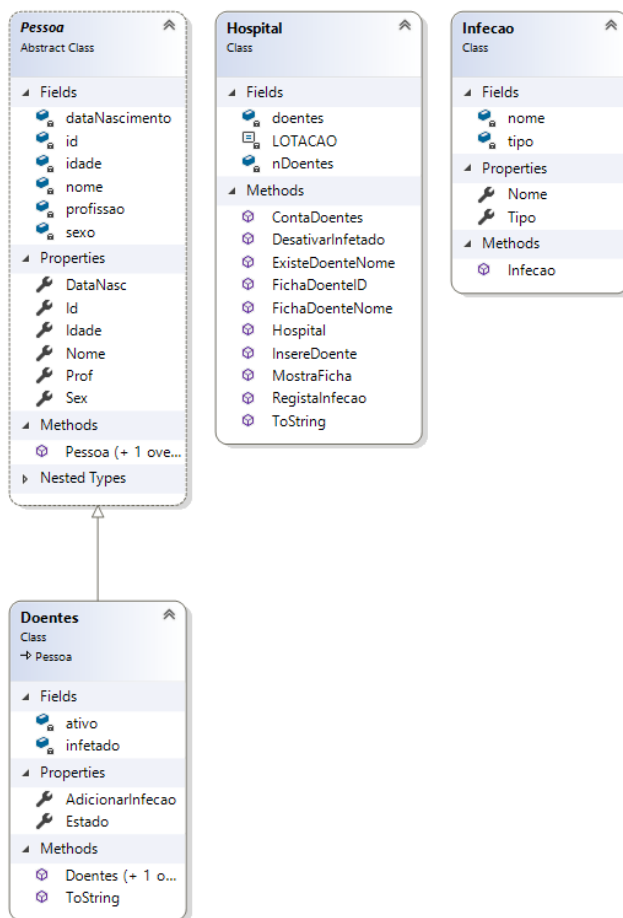


Figura 2: Diagrama de Classes

## 5.2 Estrutura das Classes

Cada classe tem a sua estrutura, olhando para a Figura 2 podemos ver nos **Fields** as estruturas das classes:

- A classe pessoa é constituída por um nome, data nascimento, id, idade, profissao e sexo.
- A classe doentes está ligada á classe pessoa e utiliza todos os seus fields e acrescenta mais dois , que são o estado e ativo, ou seja o field ativo mostra se ainda se encontra doente e o outro field corresponde á sua doença.
- A classe hospital é constituída pela lotação, número de doentes e o field doentes, que consiste nas fichas de todos os pacientes.
- A classe infeção é constituída pelo nome e tipo da doença, por exemplo : tipo - virus , nome - Varicela.

## 6 Exemplo de Execução

De seguida iremos demonstrar um exemplo da execução deste programa.

```
Nome      : Ricardo
Idade     : 32
ID        : 1
Data nascimento : 26/04/2020 00:00:00
Sexo      : Masculino
Profissao : Advogado
Tipo de doença : Bacteriano
Infetado  : True
nome      : Salmonela

Nome      : Pedro
Idade     : 45
ID        : 2
Data nascimento : 26/04/2020 00:00:00
Sexo      : Masculino
Profissao : Medico
Tipo de doença : Cancro
Infetado  : True
nome      : tumor
```

Figura 3: Exemplo de Execução

## 7 Problemas encontrados

Nesta primeira fase foram poucos os problemas encontrados, incidindo estes na criação das classes mas, com a ajudado material disponibilizado no moodle, conseguimos facilmente implementa-las no nosso programa.

## 8 Conclusão

Este trabalho permitiu-nos aprofundar os conhecimentos obtidos tanto nas aulas teóricas e aplicá-los a nível prático. No geral, consideramos satisfatórios os resultados apresentados para esta fase, pois apesar de o programa não ter muitas funções pensamos que as que temos estão bem implementadas. Em jeito de conclusão, nas fases seguintes, esperamos , como sempre , uma otimização a nível da estruturação do código e acrescentar mais funções tais como um sistema protegido com password, utilização de ficheiros e menus.