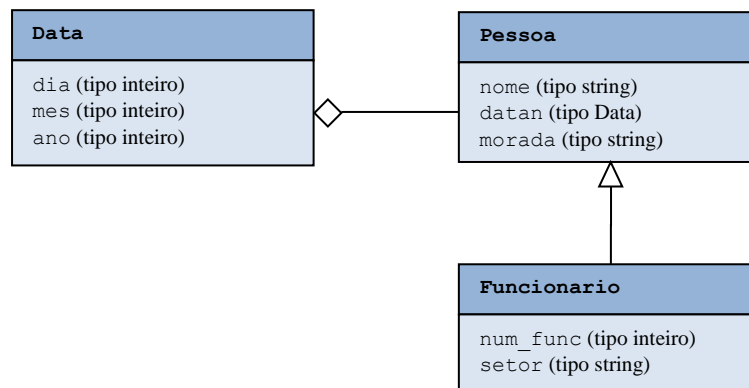


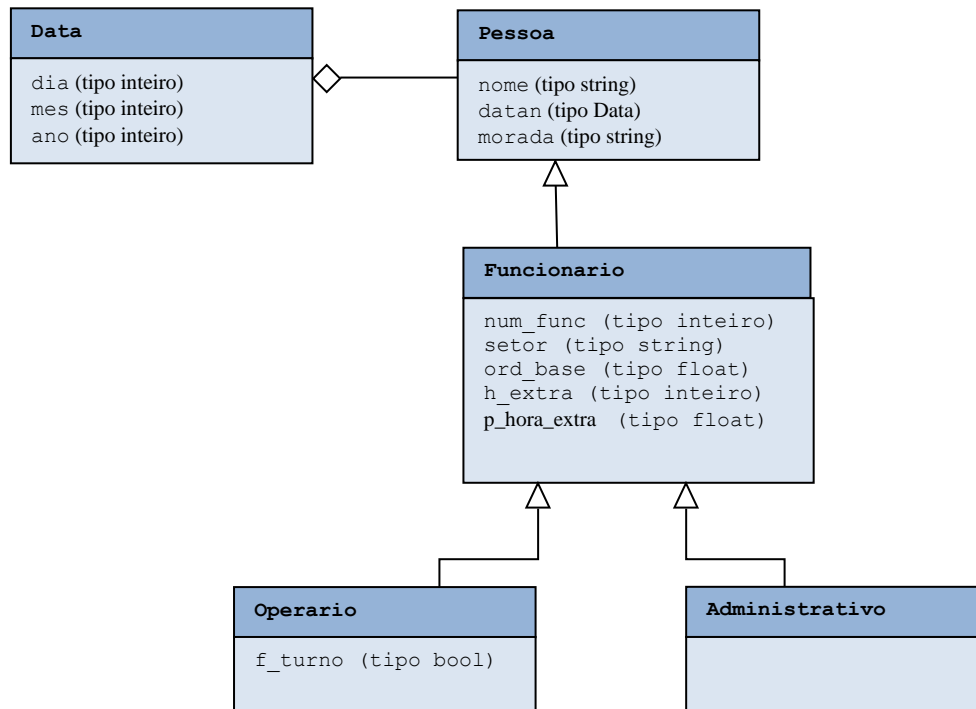
CADERNO DE EXERCÍCIOS 02

5. Considere que uma fábrica de automóveis pretende desenvolver uma aplicação para fazer a gestão dos seus funcionários. São modeladas 3 classes: *Data*, *Pessoa* e *Funcionario*.



- 5.1. Defina a classe `Funcionario`, de acordo com o diagrama apresentado.
- 5.2. Implemente os construtores por defeito e com parâmetros da classe `Funcionario`.
- 5.3. Implemente os métodos assessores da classe.
- 5.4. Num programa, defina 2 objectos do tipo `Funcionario` com os seguintes atributos:
 - a) Joca Gaio; 20/9/1987; Rua da direita n 2; 25023; Pintura.
 - b) Ana Rola; 4/2/1990; Rua da esquerda n 3; 25024; Comercial.
- 5.5. Mostre no ecrã os dados dos funcionários criados na alínea anterior.
- 5.6. Atualize a morada do "Joca Gaio" para "Rua do meio n 4" e o setor para "Ferragem".
- 5.7. Na data de nascimento da "Ana Rola", altere o dia para "23".
- 5.8. Implemente a sobrecarga do método `>>` para ler a informação de um funcionário, inserida pelo teclado.
- 5.9. Defina mais 2 objectos do tipo `Funcionario` e leia os seus dados através do teclado.
- 5.10. Implemente o método `SaveFile()` para guardar os dados de um funcionário em ficheiro.
- 5.11. No programa principal, crie um ficheiro, cujo nome seja lido através do teclado e guarde a informação dos 4 funcionários, um por linha, com os campos separados por `' '`.
- 5.12. Implemente o método `ReadFile()` para ler os dados de um funcionário a partir de um ficheiro.
- 5.13. No programa principal, leia do ficheiro, criado na alínea 5.10, os dados de todos os funcionários para um vector de objectos.
- 5.14. Mostre, no ecrã, os dados de todos os Funcionários, ordenados pela data de nascimento.

6. Na fábrica existem 2 tipos de funcionários: Operários e Administrativos. Todos os funcionários têm um ordenado base (*ord_base*) ao qual podem ser acrescentados complementos de acordo com o seu trabalho. As horas de trabalho extraordinário (*h_extra*) realizadas durante o mês são pagas como complementos, ao valor de 10€/hora para os Administrativos e de 8€/hora para os Operários. Além disso, os Operários podem optar por trabalhar por turnos e nesse caso recebem sempre um complemento de 25%, relativamente ao ordenado base.



- 6.1. Faça as devidas alterações à classe *Funcionario*, acrescentando um método virtual puro *Calcula_ordenado()*, tornando-a numa classe abstrata. Altere em conformidade os construtores com parâmetros da classe.
- 6.2. Defina as classes *Operario* e *Administrativo* de acordo com o seguinte diagrama.
- 6.3. Implemente os métodos construtores e de acesso em ambas as classes derivadas.
- 6.4. Faça a redefinição do método *Calcula_ordenado()* nas duas classes derivadas.
- 6.5. Desenvolva uma aplicação que possua um menu com as seguintes opções:
 - a) Leitura do ficheiro "Funcionario.txt" (ver exemplo), com os dados de *N* Operários e Administrativos, para um vector de apontadores. O primeiro caractere da linha indica se é um Operário (O) ou Administrativo (A).

```

O:Jonas Culatra; 20/9/1987; Rua da direita n 2; 25023; Pintura;1100;20;sim;
A:Joni Rato; 4/2/1990; Rua da esquerda n 3; 25024; Comercial;900;0;
O:Luis Carlos; 1/12/1985; Rua Central n22;10000; Ferragem; 1100;0;nao;
...
EOF
  
```

- b) Calcular e apresentar o ordenado de todos os funcionários.
- c) Pesquisar um funcionário pelo seu número.
- d) Alterar o ordenado base de um funcionário, identificado pelo seu número.
- e) Adicionar ao vector um *Operario* ou *Administrativo*, à escolha do utilizador.
- f) Apresentar o nome dos operários que trabalham por turnos.
- g) Eliminar do vetor um funcionário, identificado pelo número.
- h) Guardar, num ficheiro "**Administrativos.txt**", os dados dos Administrativos.
- i) Guardar, num ficheiro "**Operarios.txt**", os dados dos Operários.