

Parte Prática

[illegible]

Foi contactado pela UTAD Banking para desenvolver uma aplicação de gestão de contas bancárias, onde na primeira fase do projeto, tem com objetivos disponibilizar um conjunto de serviços mínimos aos seus clientes. Nesta primeira fase os utilizadores poderão criar **Conta** com apenas um **Titular** e efetuar apenas as **operações** de débito (levantamento) ou crédito (depósito).

O **Titular** é a representação do cliente, onde se regista o nome do cliente, morada e número cartão de cidadão.

1. Defina a classe **Titular** e implemente um construtor por parâmetros.

2. Implemente o método **Show()** para apresentar a informação do Titular.

Nome:	André Sousa
Morada:	Vila Real, PT

A **Conta** tem associado um **Titular**, possui também o saldo atual e a data da abertura da conta. Assuma a existência da Classe **Data** como implementada nas aulas práticas (Anexo A).

***Nota:** Identifique adequadamente a relação entre as classes.*

3. Defina a classe **Conta** e implemente um construtor por parâmetros que permita inicializar todos os seus atributos.

4. Efetue a sobrecarga do **operador <<** de forma a apresentar a informação de conta no seguinte formato na consola:

Nome:	André Sousa
Morada:	Vila Real, PT
Cliente desde:	2018-10-31
Saldo Atual:	500€

5. Implemente a sobrecarga do **operador ==** que compara se duas contas são iguais, isto é, se o titular da conta é o mesmo.

6. Implemente os métodos de **Deposito(int valor)** e **Levantamento(int valor)**, assegurando-se com os procedimentos que ache necessários de forma a garantir que saldo da conta nunca fica negativo.

7. Implemente um método **Savefile(int valor)** para acrescentar ao ficheiro “*movimentosConta.txt*” a informação relativa a todos os movimentos realizados (Depósitos/Levantamentos), com a seguinte estrutura:

Nota: O valor é positivo ou negativo conforme se trate de um movimento de depósito ou levantamento, respetivamente.

DATA;NUMERO;VALOR

2018-11-06;24328;500;

2018-11-06;24328;-100;

8. Faça um programa que crie uma conta na data de hoje para o cliente André Sousa, com o número 12324328, com morada em Vila Real, PT e com o saldo inicial de conta de 600€. Crie outra conta para o cliente Vítor Filipe com o número 145544545, também residente na mesma localidade.
- a. Apresente a informação de ambas as contas no ecrã;
 - b. Efetue uma transferência da conta do André para a conta do Vítor no valor de 174 € e apresente no ecrã o estado final das contas.

9. Implemente a sobrecarga do **operador +** que permite efetuar a união de duas contas se estas forem do mesmo titular. Nesta operação o saldo de ambas é somado, e é mantida a data da conta mais antiga. Caso o titular das contas não seja o mesmo, deve ser retornada a própria conta, sem alterações.

Anexo A:

```
#include <iostream>
#include <fstream>
using namespace std;
class Data
{
private:
    int ano, mes, dia;
public:
    Data();
    Data(int _d, int _m, int _a);
    ~Data();
    void SetDia(int _d) { dia = _d; }
    void SetMes(int _m) { mes = _m; }
    void SetAno(int _a) { ano = _a; }
```

```
int GetDia(void) { return dia; }
int GetMes(void) { return mes; }
int GetAno(void) { return ano; }
void Show(void);
bool operator == ( Data data) const;
bool operator < (const Data data) const;
bool operator > (const Data data) const;
friend ostream & operator << (ostream &os, Data data);
friend istream & operator >> (istream &is, Data &data);
void SaveFile(ofstream& os);
void ReadFile(istream& is);
};
```