

Agência de Viagens “NiceHolidays”

INTRODUÇÃO

A agência de viagens **NiceHolidays** comercializa diversos tipos de pacotes de viagens para diferentes tipos de público-alvo. A empresa pretende desenvolver um programa informático para gerir a sua atividade no setor do turismo. A sua atividade envolve a gestão de **clientes** e de **pacotes turísticos**.

Uma **agência** é caracterizada pelos seguintes atributos: nome, número de identificação fiscal (NIF), morada, URL do seu *web site*, lista de pacotes turísticos que comercializa, e lista de clientes.

Cada **cliente** é caracterizado por: nome, ~~idade~~, NIF, número de pessoas no agregado familiar, morada, lista de pacotes turísticos comprados, e total de compras efetuadas.

Um **pacote turístico** inclui: um identificador único, o(s) local(ais) turístico(s) a visitar, a data de início, a data de fim, o preço por pessoa e o número máximo de pessoas que podem inscrever-se no pacote.

Uma **morada** inclui: o nome da rua, o número da porta, o número do andar ("- se não aplicável), o código postal e a localidade.

Uma **data** é representada por: ano, mês e dia.

O objetivo último do trabalho prático é desenvolver um programa que permita a uma agência de viagens: i) gerir pacotes de viagens (adicionar, alterar ou eliminar); ii) gerir clientes (adicionar novos clientes, vender-lhes um pacote turístico; ou eliminar clientes). Neste primeiro trabalho serão apenas desenvolvidas as funcionalidades básicas. Estas funcionalidades serão complementadas no segundo trabalho, onde o desenvolvimento da aplicação completa será concluído.

OBJETIVOS

Este trabalho prático tem como objetivo desenvolver um programa para preparar a aplicação de gestão da agência de viagens **NiceHolidays**. O programa a desenvolver deve definir novos tipos de dados, adequados à aplicação, criar as estruturas de dados adequadas, ler/gravar informação de/para ficheiros. Não é objetivo desenvolver um programa completo que faça a gestão da agência; esse será o objetivo do trabalho prático nº 2, usando um conjunto de regras que serão indicadas oportunamente.

Ao realizar este trabalho, os estudantes terão oportunidade de aplicar os conhecimentos de programação em C++ adquiridos até ao momento, nomeadamente, quanto a:

- desenvolvimento de interfaces simples;
- tratamento de entradas do teclado inválidas;
- formatação de saídas;
- leitura e escrita em ficheiros de texto;
- uso de estruturas de controlo de um programa (seleção e repetição);
- uso de diversos tipos de estruturas de dados (*strings*, *arrays/vectors*, *structs*, *files*);
- definição de novos tipos de dados apropriados para a aplicação ;
- uso de funções, com passagem de parâmetros por valor e por referência.

DESENVOLVIMENTO DO PROGRAMA

Definição de novos tipos de dados

Utilizando a facilidade de definição de estruturas (**struct**) e de definição de novos tipos de dados (**typedef**), disponíveis na linguagem, comece por definir novos tipos de dados para: agência de viagens (**Agency**), cliente (**Client**), pacote turístico (**TravelPack**), morada (**Address**), e data (**Date**). Defina também as estruturas de dados necessárias para guardar os dados de todos os clientes e de todos os pacotes turísticos.

Funcionamento geral

O programa deve começar por ler o ficheiro **agency.txt** que contém: a informação estática da agência (nome, NIF, URL e morada), o nome do ficheiro onde está guardada a lista de clientes (ex: **clients.txt**) e o nome do ficheiro onde está guardada a informação dos pacotes turísticos (ex: **packs.txt**). Uma vez obtido o nome dos ficheiros onde está a informação de clientes e pacotes turísticos, estes devem ser lidos de seguida, de modo a completar a informação da agência relativa a clientes e pacotes turísticos. Cada item de informação no ficheiro **agency.txt** ocupa uma linha respeitando a seguinte ordem: nome, NIF, URL, morada, nome do ficheiro de clientes e nome do ficheiro de pacotes turísticos. Uma morada é guardada numa linha com a seguinte sintaxe: "rua / número da porta / número do andar / código postal / localidade" (ver exemplo na próxima secção).

No ficheiro de **clientes**, a informação de cada cliente é escrita a seguir à do anterior, apenas separada por uma linha contendo a *string* ":::::::::" (10 caracteres ':' consecutivos). Cada item de informação de cada cliente é escrito numa linha separada. A ordem dos itens/linhas de um cliente é: nome, ~~idade~~, NIF, número de pessoas no agregado familiar, morada, lista de identificadores dos pacotes turísticos comprados e valor total de compras efetuadas.

No ficheiro de **pacotes turísticos**, a informação de cada pacote é escrita a seguir à do pacote anterior, apenas separada por uma linha contendo a *string* ":::::::::" (10 caracteres ':' seguidos). Cada item de informação de cada pacote é escrito numa linha separada. A ordem dos itens/linhas de um pacote são: identificador único, o(s) local(ais) turístico(s) a visitar, data de início, data de fim, preço por pessoa, e número máximo de pessoas. Uma data tem o seguinte formato: ano/mês/dia.

O formato dos ficheiros será ilustrado adiante.

Os dados lidos dos ficheiros de clientes e de pacotes, no início do programa, devem ser guardados nas estruturas de dados, internas ao programa, escolhidas para o efeito. Antes de fim de cada execução do programa, o conteúdo dessas estruturas de dados deve ser escrito nos ficheiros respetivos, de forma a manter os dados atualizados.

Ficheiro de agências, de clientes e de pacotes turísticos

Os ficheiros utilizados no programa são ficheiros de texto que podem ser criados com um editor de texto simples como, por exemplo, o *Notepad* do *Windows*. O programa deve começar por perguntar ao utilizador qual o nome do ficheiro da agência a usar, terminando com um erro caso algum dos ficheiros indicados (agência, clientes e pacotes turísticos – os nomes dos dois últimos não são pedidos, estão nas 2 últimas linhas do ficheiro da agência) não exista.

Apresenta-se a seguir um possível conteúdo de um ficheiro da agência.

```
NiceHolidays
133331145
http://www.niceholidays.pt
Rua Sem Nome / 100 / - / 4400-345 / Porto
clients.txt
packs.txt
```

Exemplo inventado para a agência NiceHolidays

Cada uma das linhas do ficheiro contém a seguinte informação: o nome da agência, o NIF, o URL, a morada (elementos da morada separados pelo carácter '/'), nome do ficheiro contendo informação de clientes, e nome do ficheiro contendo informação sobre os pacotes turísticos.

Um conteúdo possível de um ficheiro de clientes é de seguida mostrado.

```
Rui Manuel
234987156
4
Rua Sem Fim / 200 / 5Esq /1200-001 / Lisboa
10 ; 36 ; 2
:::::::::
Belmiro Miguel
111987666
2
Avenida dos Grilos / 100 / - /2300-101 / Coimbra
100 ; 136 ; 20
```

Exemplo de um ficheiro de clientes com 2 clientes

Cada uma das linhas do ficheiro contém a seguinte informação para cada cliente: nome do cliente, NIF, número de pessoas no agregado familiar, morada (elementos da morada separados pelo carácter '/'), lista dos pacotes turísticos que já comprou (elementos da lista separados por ponto e vírgula). Uma linha com 10 caracteres ':' consecutivos separa um cliente do seguinte.

Um conteúdo possível de um ficheiro de pacotes turísticos é de seguida mostrado.

```
3
1
Madeira - Funchal, Porto Santo
2019/08/01
2019/08/05
300
30
10
:::::::::
-2
Veneza
2019/03/02
2019/03/05
300
24
24
:::::::::
3
Douro vinhateiro - Porto, Régua, Pinhão, Vila Real
2019/09/21
2019/09/22
100
62
12
```

Exemplo de um ficheiro de pacotes turísticos com dois pacotes

A primeira linha indica o identificador numérico único que foi atribuído ao último pacote que foi criado; segue-se a informação dos pacotes turísticos. Cada uma das linhas de um pacote turístico contém a seguinte informação: identificador numérico único do pacote (positivo se o pacote estiver disponível, negativo se a pacote não estiver disponível), principal local turístico de destino (pode ser seguido de um hífen e de uma lista dos principais locais turísticos a visitar, separados por vírgula), data de início da viagem (elementos da data separados pelo carácter '/'), data de fim da viagem (elementos da data separados pelo carácter '/'), preço por pessoa, número de lugares inicialmente disponíveis no pacote, e número de lugares vendidos. Uma linha com 10 caracteres ':' consecutivos separa um pacote turístico do seguinte. Nota: neste caso, foram criados 3 pacotes turísticos, mas o pacote número 2 já não está disponível (tendo por isso um identificador negativo).

Funcionalidades a implementar

A parte da aplicação que deve implementar neste trabalho inclui as seguintes funcionalidades:

1. Ler e guardar a informação da **agência**, dos **clientes** e dos **pacotes turísticos** armazenada em ficheiros.
2. Gerir os **clientes** e **pacotes turísticos**: criar, alterar e remover um **cliente** ou um **pacote turístico**.
3. Gerar e visualizar de modo formatado a informação de um cliente especificado.
4. Gerar e visualizar de modo formatado a informação de todos os **clientes** da agência.
5. Gerar e visualizar de modo formatado os **pacotes turísticos** disponíveis (todos, todos relativos a um destino específico, todos entre duas datas, todos os relativos a um destino específico e entre duas datas).
6. Gerar e visualizar de modo formatado os **pacotes turísticos** vendidos (a um cliente específico, a todos os clientes).
7. Efetuar a compra de um pacote turístico por um cliente.
8. Calcular e visualizar o número e o valor total de pacotes vendidos.

NOTAS SOBRE O DESENVOLVIMENTO

Funcionamento geral do programa

O funcionamento geral do programa é o seguinte:

- Ler o conteúdo dos ficheiros da agência, dos clientes e dos pacotes turísticos e parâmetros da aplicação e guardar numa estrutura de dados adequada.
- Começar por exibir um menu onde aparecem as opções que implementam as funcionalidades: gestão de clientes; gestão de pacotes turísticos; visualização de informação (sobre clientes e sobre pacotes turísticos).
- No final, gravar (se necessário) a informação da agência dos clientes e dos pacotes turísticos.

Devem ser tomadas as precauções necessárias para evitar que o programa deixe de funcionar corretamente devido a entradas inválidas do utilizador, nomeadamente, valores fora da gama admissível.

Escrita do código

Na escrita do código devem ser respeitadas as indicações dadas nas aulas, nomeadamente, em relação aos seguintes aspetos:

- Escolha das estruturas de dados mais adequadas para representar os dados do programa.
- Escolha adequada dos identificadores de tipos, variáveis e funções.
- Estrutura modular do código.
- Comentários ao código.

ENTREGA DO TRABALHO

- Criar uma pasta com o nome **TxEYYYYYYYYY**, em que **x** representa o número da turma e **YYYYYYYYYY** representa o número do estudante (sem as letras **up** ou **ei**); por exemplo, **T2E201500007**, para o estudante com o número 201500007 da turma 2) e copiar para lá o código-fonte do programa (apenas os ficheiros com a extensão .cpp e .h, se existirem). Incluir também um ficheiro ReadMe.txt (em formato de texto simples) indicando o estado de desenvolvimento do trabalho, isto é, se foram cumpridos todos os objetivos ou, caso contrário, quais os que não foram cumpridos, ou ainda que melhorias foram implementadas, se for esse o caso.
- Compactar o conteúdo desta pasta num ficheiro **TxEYYYYYYYYY.zip** ou **TxEYYYYYYYYY.rar** e submeter este ficheiro na página da disciplina de Programação, no Moodle da FEUP. Não serão aceites entregas por outras vias.
- Data limite para a entrega: **12/abr/2019** (às 23:55h).

NOTA: Apesar de o 1.o trabalho prático não ser classificado, é condição necessária para a obtenção de frequência o desenvolvimento, entrega e apresentação (quando solicitada) deste trabalho. O trabalho será realizado individualmente.