

 <p>UNIVERSIDADE DE COIMBRA FACULDADE DE CIÊNCIAS E TECNOLOGIA Departamento de Engenharia Informática</p>	<h2>Princípios de Programação Procedimental</h2> <p>Projeto 2018/19 Planeamento de Viagens no DEI</p> <p>Data de Entrega: 24 de maio de 2019</p>
<p><u>Nota:</u> A fraude denota uma grave falta de ética e constitui um comportamento não admissível num estudante do ensino superior e futuro profissional. Qualquer tentativa de fraude pode levar à reprovação na disciplina tanto do facilitador como do prevaricador, para além de outras consequências previstas na lei.</p>	

Objetivo

Pretende-se com este projeto desenvolver uma aplicação de suporte a organização de visitas de estudo.

Competências

1. Escrita de programas em C. Domínio do ambiente de desenvolvimento.
2. Escrita de código corretamente formatado e indentado.
3. Acesso a ficheiros.
4. Definição de novos tipos de dados.
5. Domínio de estruturas de dados dinâmicas.
6. Utilização de listas ligadas.
7. Utilização de soluções eficientes de ordenamento.

Descrição do Problema

Os alunos do DEI pretendem organizar uma viagem de estudo e querem desenvolver uma aplicação de suporte ao planeamento da viagem. A viagem deve contemplar a deslocação a 3 locais, e a visita de 3 pontos de interesse (PDI) em cada local. Os locais a visitar são caracterizados pelo nome da cidade onde estão situados. Cada local pode ter 3 ou mais PDIs, que podem ser: museus, parques, bares, etc. Todos os PDIs têm um nome, uma descrição e um horário de funcionamento.

A aplicação deve ajudar a escolher as opções de viagem que satisfazem as preferências dos alunos e deve mostrar quais são os PDIs mais populares. A interação com o utilizador deverá ser realizada através da linha de comandos. Os dados iniciais da aplicação deverão constar de um ficheiro com a lista dos locais e seus PDIs, sendo que estes dados não se alteram durante a execução do programa. Este ficheiro deve conter pelo menos 20 locais, cada um com um mínimo de 3 PDIs diferentes. Após o

registo do primeiro utilizador, todos os restantes dados devem ser guardados também em ficheiro e carregados sempre que a aplicação for iniciada.

A aplicação deve permitir ao utilizador escolher os locais e Pdl's preferidos, além de um único Pdl "hot", que este não quer perder. Na prática, a aplicação deve permitir realizar as seguintes operações:

1. Registo dos utilizadores, incluindo nome, morada, data de nascimento e telefone. Estes dados podem ser alterados mais tarde.
2. Listagem por ordem alfabética dos locais e Pdl's existentes. Esta ordem é feita primeiro pelos locais e depois pelos Pdl's de cada local.
3. Adição e remoção de preferências do utilizador relativas a locais e Pdl's, bem como do seu Pdl "hot", a partir das listas disponibilizadas em 2. O utilizador só pode ter, em cada momento, um máximo de 3 locais preferidos, mas pode ter tantos Pdl's preferidos quantos quiser, independentemente dos locais a que pertencem.
4. Listagem dos locais e Pdl's mais populares, considerando as preferências de todos os utilizadores registados até ao momento. Deve listar os locais por ordem de interesse dos utilizadores e, dentro de cada local, deve ordenar também pelos Pdl's mais populares.
5. Construir de forma automática uma viagem baseada nos três locais preferidos do utilizador. Esta opção só deve estar disponível **quando o utilizador tiver três locais preferidos definidos**. Na construção da viagem, deve seleccionar 3 Pdl's em cada local, dando prioridade ao Pdl "hot", se estiver nesse local, seguido dos Pdl's preferidos do utilizador, ordenados por popularidade decrescente, e dos restantes Pdl's também ordenados por popularidade decrescente. A viagem deverá ser mostrada ao utilizador, de forma a que ele possa visualizar todos os locais e Pdl's incluídos.
6. Avaliar a viagem construída/visualizada no ponto anterior, de acordo com a sua **taxa de popularidade**, que é definida por uma média dos seguintes fatores:
 - % de utilizadores que tem pelo menos 1 local favorito entre os incluídos na viagem;
 - % de utilizadores cujo Pdl "hot" está incluído nesta viagem;
 - % das preferências de Pdl: total de pontos de preferência dos Pdl's incluídos a dividir pelo total dos pontos de preferência de todos os Pdl's existentes.

Implementação

A aplicação deve ser implementada na linguagem C e deverá ser baseada em listas ligadas. Mais concretamente, deverá usar listas ligadas em memória para manter os dados dos locais e respetivos Pdl's, assim como dos utilizadores. As relações entre estas deverão depois ser mantidas com novas listas ligadas de apontadores. Por exemplo, cada pessoa poderá ter uma lista ligada de apontadores para os seus locais favoritos, enquanto que cada local também pode apontar para pessoas. A mesma estratégia se poderá aplicar aos Pdl's. Algumas destas listas deverão ser ordenadas, por exemplo, a lista de locais, de forma a permitir uma visualização rápida. Note que deverá ter o cuidado de manter estas listas num estado consistente, mesmo quando forem alteradas as preferências do utilizador do programa.

Utilize ficheiros para armazenar os dados sempre que sair da aplicação e para os recuperar, sempre que reentrar. Como os locais e pontos de interesse não se alteram, deve usar outro(s) ficheiro(s) para

os restantes dados. Deve utilizar ficheiros de texto, para permitir a sua leitura, quer por si, quer pelo seu professor. É importante que compreenda que a implementação que vai fazer desta aplicação tem um objetivo pedagógico de o introduzir na linguagem C e de lhe permitir o domínio da programação com apontadores. Numa aplicação real faria sempre recurso de outras tecnologias, como bases de dados e de outras estruturas de dados que terá oportunidade de conhecer noutras disciplinas mais avançadas do curso.

Composição dos grupos

O trabalho deverá ser realizado por grupos de 2 elementos pertencentes a turmas do mesmo Professor.

Material a entregar

Cada grupo deve entregar obrigatoriamente:

1. Upload de zipFile no InforEstudante com:
 - Todo o código fonte (.c e .h).
 - Ficheiros de dados para teste.
 - Relatório (em formato pdf).
2. O relatório deve incluir os seguintes aspetos:
 - Esquema com as estruturas de dados utilizadas.
 - Estrutura geral do programa.
 - Estrutura dos ficheiros de texto (que dados são armazenados em cada ficheiro e como).
 - Breve explicação de como o programa se executa.

Defesa final do trabalho

O trabalho deverá ser defendido através de uma prova oral com todos os elementos do grupo. Os estudantes que não comparecerem à defesa do trabalho terão a classificação de **zero** no projeto.

A nota do projeto será atribuída em função da avaliação do trabalho entregue e da sua defesa.

Apesar do trabalho ser de grupo, as notas são individuais, podendo ser atribuídas notas diferentes a cada elemento do grupo, sempre que tal se justifique.

As defesas decorrerão após a entrega dos trabalhos, em data a anunciar. Será colocado no *inforestudante* um mapa de defesas onde os grupos se devem inscrever.