Programação

LEI; LEI-PL; LEI-CE - 2022/23

Trabalho Prático

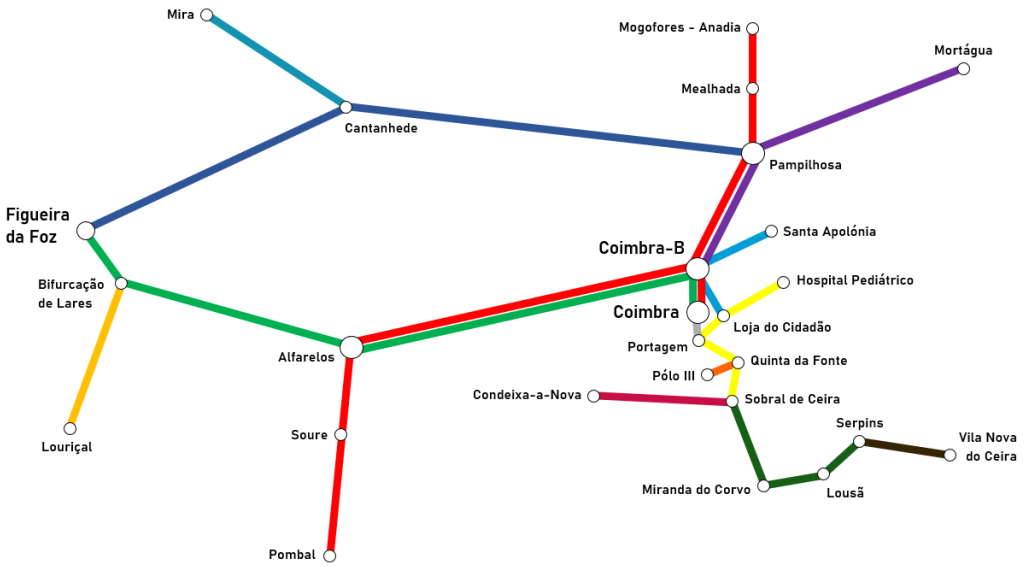
Metro Mondego

Notas prévias:

- O enunciado é possivelmente vago, genérico e incompleto em alguns pontos. O que se pretende é que os alunos avaliem as opções existentes e escolham a que considerarem mais apropriada para cada uma das situações com que se depararem. Todas as escolhas devem ser referidas e justificadas no relatório.
- O programa deve ser implementado em C *standard*, *i.e.,* não deve ter instruções que o tornem específico para um determinado ambiente/plataforma de desenvolvimento. Deverá ser respeitada a norma C99.
- O programa entregue deve ter uma interface simples e amigável, indicando o que pode ser feito em cada situação. Não são valorizados programas com interfaces gráficas e/ou utilização de bibliotecas que não sejam *standard*, por exemplo, para usar cores ou posicionar o cursor no ecrã.
- Deve distribuir o código fonte por vários ficheiros. Deverão existir, no mínimo, três ficheiros com código fonte.
- Deverá utilizar *header files* para gerir as dependências entre os vários ficheiros com código fonte.
- Cada ficheiro deve ter a identificação do aluno (nome completo e número), em comentário, nas linhas iniciais do ficheiro.

1. Introdução

A empresa que gere o sistema de mobilidade do Metro Mondego pretende um programa que efetue a gestão das várias linhas existentes. Este programa deverá manter informação atualizada sobre as linhas que constituem o sistema e sugerir percursos entre 2 paragens.



Cada linha do sistema Metro Mondego é constituída por várias paragens. Em cada paragem podem entrar e sair pessoas. As linhas podem ser percorridas nos 2 sentidos. Pode considerar que as linhas não são circulares e nenhuma delas passa 2 vezes pela mesma paragem.

Exemplo

A *Linha da Baixa* é constituído por 5 paragens, entre o Parque da Cidade e Coimbra-B. Pode ser percorrida nos seguintes sentidos:

Parque da Cidade ⇒ Portagem ⇒ Loja do Cidadão ⇒ Casa do Sal ⇒ Coimbra-B

Coimbra-B ⇒ Casa do Sal ⇒ Loja do Cidadão ⇒ Portagem ⇒ Parque da Cidade

O sistema é constituído por diversas linhas que se cruzam em algumas paragens. Nesses locais, um passageiro pode sair de uma das linhas e entrar numa outra.

2. Programa a Implementar

Pretende-se que desenvolva um programa em linguagem C que permita efetuar a gestão do sistema de mobilidade do Metro Mondego As funcionalidades previstas são descritas nos pontos seguintes.

2.1 Paragens

As paragens pertencentes ao sistema são identificadas por um nome e por um código alfanumérico com 4 caracteres. Tanto o nome, como o código alfanumérico são únicos, ou seja, não podem existir 2 paragens com o mesmo nome e/ou o mesmo código alfanumérico. Pode adicionar outros campos, caso julgue conveniente (por exemplo, a quantas linhas pertence uma determinada paragem ou outra informação relevante). Durante a execução do programa, esta informação está armazenada num **array de estruturas dinâmico**.

2.1.1 Operações sobre Paragens

- **Registar Paragem**: Podem ser adicionadas novas paragens ao sistema. O nome é introduzido pelo utilizador, mas o código alfanumérico único deve ser gerado automaticamente pelo programa (este campo não é especificado pelo utilizador). Quando uma nova paragem é adicionada, não fica associada a nenhuma linha. Isso será efetuado mais tarde (ver ponto 2.2.1).

- **Eliminar Paragem**: Pode ser eliminada uma paragem do sistema. O código da paragem a eliminar é introduzido pelo utilizador. Só podem ser eliminadas paragens que, nessa altura, não façam parte de nenhuma linha.

- **Visualizar Paragens:** A lista completa de paragens existentes no sistema pode ser apresentada na consola.

2.2 Gestão de Linhas

A informação completa sobre as linhas do metro deve ser mantida numa estrutura dinâmica do tipo **lista ligada**.

2.2.1 Operações sobre Linhas

- **Adicionar Linhas**: Podem ser adicionadas novas linhas ao sistema de mobilidade. A adição pode ser efetuada a partir de informação introduzida pelo utilizador ou lendo esses dados de um ficheiro de texto[1]. Toda a informação necessária para criar a linha deve ser especificada, incluindo o seu nome e as várias paragens. Não é possível ter duas linhas com o mesmo nome, nem incluir na linha paragens não registadas no sistema.

- **Atualizar Linha**: A sequência de paragens de uma linha pode ser atualizada. A atualização pode consistir na adição ou eliminação de uma ou mais paragens.

[1] O formato do ficheiro de texto é descrito no ponto 3.

- **Visualizar Linhas**: O programa deve permitir a apresentação completa de todas as linhas existentes no sistema. Também deve ser possível mostrar as linhas que passem numa determinada paragem.

2.3 Cálculo de Percursos

Nesta funcionalidade, o programa deverá apresentar uma listagem com todos os percursos que liguem 2 paragens. O utilizador indica os nomes do ponto de partida e do ponto de chegada e o programa mostra quais as linhas que permitem fazer a ligação. Deve ser indicado todo o percurso, isto é, todas as paragens entre o ponto de partida e o ponto de chegada.
A listagem deve considerar 2 possibilidades:

i. Percurso efetuado numa única linha
ii. Percurso com uma mudança de linha[2]

2.4 Armazenamento em Ficheiro

Imediatamente antes de terminar a execução, o programa deve guardar a informação das paragens e das linhas num **ficheiro binário**. Esta informação deverá permitir reconstruir as estruturas dinâmicas (array dinâmico e lista ligada) quando o programa retomar a execução.

3. Ficheiro de Texto

A informação sobre uma nova linha a adicionar ao sistema pode ser especificada através de um ficheiro de texto. Cada ficheiro de texto só pode conter informação sobre uma linha.

```
Linha da Baixa
Parque da Cidade  #  P011
Portagem  #  P123
Loja do Cidadao  #  Q998
Casa do Sal  #  F554
Coimbra-B  #  H123
```

O formato é o seguinte:

i. - Na primeira linha do ficheiro surge o nome da nova linha (i.e., o nome do percurso) do metro
ii. - Nas linhas seguintes do ficheiro surgem o nome e o identificador alfanumérico das paragens (1 paragem por linha).

O ficheiro ao lado ilustra a informação necessária para construir a linha descrita no exemplo do ponto 1.

Parque da Cidade ⇒ Portagem ⇒ Loja do Cidadão ⇒ Casa do Sal ⇒ Coimbra-B

Coimbra-B ⇒ Casa do Sal ⇒ Loja do Cidadão ⇒ Portagem ⇒ Parque da Cidade

O caracter '#' separa o nome da paragem do seu identificador alfanumérico. Pode assumir que este caracter nunca faz parte do nome ou do identificador e atua apenas como separador.

[2] Os percursos entre o ponto inicial e o final poderão ter, no máximo, um transbordo

4. Normas para a realização do trabalho

O trabalho deve ser **realizado individualmente**. O trabalho só pode ser entregue uma vez e a nota obtida é válida em todas as épocas de avaliação do ano letivo 2022/23.
Não existirá um novo trabalho ou uma substituição desta componente de avaliação para a época especial de setembro.

Data limite para entrega do trabalho prático: **23.59** do dia **11 de junho de 2023.**

Material a entregar:

- Entregar através do Moodle um ficheiro compactado em formato ZIP, contendo o relatório, o código fonte comentado (apenas os ficheiros .c e .h) e os ficheiros de dados necessários para o funcionamento do programa.
- O nome do ficheiro ZIP deve obrigatoriamente ter o seguinte formato: `Prog_Nome_NumAluno.zip`, em que *Nome* e *NumAluno* identificam, respetivamente, o nome e número do do aluno (exemplo: *Prog_AnaSilva_123456789.zip*)

Defesa:
Os trabalhos serão sujeitos a **defesa obrigatória,** em data e formato a anunciar. As defesas poderão incluir:

i) Demonstração do funcionamento do programa
ii) Explicação detalhada do código
iii) Implementação de alterações / novas funcionalidades

Relatório

Deve ser entregue um relatório contemplando os seguintes pontos:

- Descrição genérica da organização do programa
- Identificação do ambiente de desenvolvimento utilizado durante a implementação
- Apresentação das estruturas dinâmicas implementadas e da organização dos ficheiros utilizados pelo programa, justificando as escolhas feitas

Avaliação

A cotação do trabalho é de **6 valores**.

Esta componente da avaliação não tem nota mínima.

A deteção de plágio parcial ou total implica a anulação imediata de todos os trabalhos envolvidos.

Critérios de Avaliação para as Funcionalidades Implementadas

- Definição das estruturas de dados
- Correção das funcionalidades implementadas
- Manipulação de estruturas dinâmicas
- Manipulação de ficheiros
- Simplicidade/funcionalidade da interface com o utilizador