ROTAS AÉREAS ENTRE CIDADES MARCIANAS

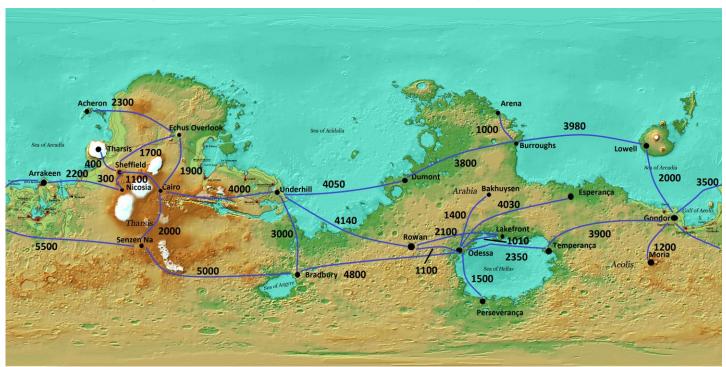
Depois da colonização e terraformação de Marte, em 2187 a Federação Marciana declarou-se independente da Terra e proporcionou aos seus habitantes uma malha aeroviária realizada por dirigíveis ligando os principais centros populacionais do planeta.

A Federação deseja fornecer aos seus usuários um aplicativo que permita verificar os caminhos entre essas principais cidades, através de viagens aéreas.

Para tanto, um arquivo texto contendo nomes de cidades, distância entre elas, tempo de percurso e preço da passagem é fornecido. Esse arquivo se chama CaminhosEntreCidadesMarte.txt.

Há também um arquivo texto chamado CidadesMarte.txt, com um número sequencial associado a cada cidade, o nome da cidade e a coordenada cartesiana (x, y) proporcional às dimensões de um mapa de Marte, onde a cidade deve ser localizada no mapa.

Futuramente, deseja-se que o aplicativo encontre todos os caminhos entre as cidades A e B indicadas e relacione o mais vantajoso, de acordo com o critério de menor distância percorrida.



O arquivo de cidades deve ser armazenado em uma lista duplamente ligada, ordenada pelo código das cidades, para busca dos códigos das cidades e de suas coordenadas para posicionamento dos extremos das linhas retas que determinam os caminhos entre cidades que façam parte dos percursos selecionados. A busca deve ser feita pelo número de identificação da cidade.

O arquivo "Mapa Marte sem Rotas.jpg" apresenta o mapa acima sem as ligações do grafo.

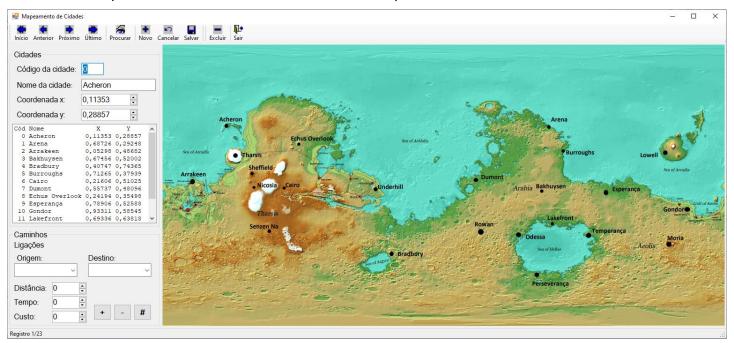
O arquivo "mars_political_map_by_axiaterraartunion_d4vfxdf-pre.jpg" é o mapa original. Ele possui 4096 x 2048 pixels e as coordenadas presentes no arquivo CidadesMarte.txt foram determinadas proporcionalmente de acordo com essas dimensões.

Obviamente, o mapa no seu tamanho original não caberá na tela. Portanto, permita que o mapa seja armazenado num componente PictureBox que se ajuste ao tamanho da tela e lembre-se que isso mudará as coordenadas de exibição de cada cidade no mapa **proporcionalmente** à mudança da altura y e largura x do mapa apresentado na tela, numa proporção entre a largura e a altura da tela com a coordenada (X, Y) original da cidade.

O programa deverá implementar os métodos da Classe **ListaDupla** solicitados pela Interface **IDados**, usando uma lista duplamente ligada como estrutura de armazenamento e recuperação de dados. Essa interface define métodos que deverão ser usados no formulário para navegar na lista duplamente ligada de cidades, incluir, excluir, exibir, pesquisar, listar dentre outras funcionalidades, de maneira semelhante à manutenção de dados que estudamos no 1º semestre.

Esse programa é a primeira parte de um projeto mais completo que, ao final deste semestre, buscará caminhos entre as cidades que forem cadastradas nos arquivos.

Dessa forma, a parte de "Caminhos" ainda não será feita, apenas a de "Cidades".



No início da execução do programa, um OpenFileDialog deve ser exibido para que o nome do arquivo com os dados seja definido pelo usuário e o programa possa ler esse arquivo e armazenar seus registos na lista duplamente ligada.

Após isso, o programa deverá mostrar cada cidade no mapa, usando um círculo preenchido e o nome da cidade para mostrar onde a cidade se encontra.

O mapa poderá variar de tamanho caso você aumente ou diminua o tamanho do formulário. Assim, como as coordenadas X e Y são proporcionais, para determinar a o pixel (x, y) exato da cidade no mapa que está sendo exibido, a largura do mapa deve ser multiplicada pela coordenada X da cidade e a altura do mapa deve ser multiplicada pela coordenada Y da cidade. Por exemplo, se o mapa estiver sendo exibido com tamanho de 1200 x 600 pixels (largura x altura), para a cidade de Acheron do exemplo acima as coordenadas onde o círculo deve ser exibido nesse mapa será (136, 173). Você pode escrever o nome da cidade abaixo do círculo.

A atualização do PictureBox para que os nomes de cidades sejam exibidos é feita no evento Paint do PictureBox.

Após a exibição dos círculos e nomes das cidades no mapa, o programa deve entrar em modo de navegação (SituacaoAtual = navegando) e o ponteiro Atual da lista deve ser posicionado no primeiro nó. O nó exibido deve ser o nó atual, se houver.

Conforme os botões de navegação forem pressionados, o ponteiro Atual deve ser movido dentro da lista duplamente ligada, através dos métodos de navegação que você codificará ao implementar os métodos já especificados na classe ListaDupla.

Quando o programa for finalizado, deve-se gravar os dados armazenados na lista duplamente ligada. Essa gravação será feita no arquivo selecionado pelo usuário no início da execução do programa.

Descrição dos arquivos

CidadesMarte.txt

IdCidade – inteiro, 3 posições NomeCidade – string, 15 posições CoordenadaX – real, 7 posições CoordenadaY – real, 7 posições

CaminhoEntreCidadesMarte.txt

idCidadeOrigem – inteiro, 3 posições idCidadeDestino – inteiro, 3 posições distancia – inteiro, 5 posições tempo – inteiro, 4 posições custo – inteiro, 5 posições

IMPORTANTE

- Trabalho feito em dupla;
- Desenvolver em C# no Visual Studio;
- Comentar adequadamente o programa e o código programado;
- Nomear os identificadores de forma adequada;
- No início dos arquivos fonte, digitar comentário com os RAs e nomes dos alunos;
- Relatório de desenvolvimento deve ser feito num arquivo cujo nome é:
 RA1_RA2_RelatorioProjeto1ED.PDF (exemplo: 21101_21192_RelatorioProjeto1ED.pdf).
 Deve conter imagens da execução da busca de caminhos;
- O relatório deve ser entregue em formato PDF;
- Entrega: 10/05/2019 (31 dias), pelo Google Classroom
- Material a ser entregue: pasta do projeto, arquivos de dados e PDF compactados em um único arquivo, cujo nome será RA1_RA2_Projeto3ED.rar (21101_21192_Projeto1ED.rar, por exemplo).

Fontes de informação:

https://www.deviantart.com/axiaterraartunion/art/Mars-Political-Map-294675891 http://www.teses.usp.br/teses/disponiveis/8/8147/tde-21102009-170528/pt-br.php https://en.wikipedia.org/wiki/Mars_trilogy

