

# Relatório do projeto de Estrutura de Dados II

Integrantes da dupla:

Gustavo Henrique de Meira – 18015

Pedro Gomes Moreira – 18174

## Linguagem

Java (Android Studio)

## Introdução

O projeto se trata de um aplicativo mobile que se baseia no algoritmo de Dijkstra para obter o melhor caminho entre uma cidade e outra de acordo com tempo ou distância, possibilidade escolhida pelo usuário antes de efetuar a busca.

Além disso, haverá a representação gráfica do percurso na tela, e a possibilidade de inserção de novas cidades no mapa, tais como novos caminhos.

A proposta de trabalho deu-nos oportunidade de testar nossos conhecimentos acerca das seguintes estruturas de dados: lista ligada, pilha, tabela hash e grafo.

## Desenvolvimento

- 23/10 às 09h10min: início do desenvolvimento do projeto. Criamos o repositório no GitHub e optamos pelo Xamarin – programação Android no Visual Studio, na linguagem C#. Adição das classes *BucketHash*, *No*, *Lista*, *IStack*, *Pilha*, *PilhaVaziaException*, *Grafo* e *Vertice*, que desenvolvemos anteriormente na matéria.

- 05/11 às 10h00min: adaptação das classes *BucketHash* e *Lista* para o projeto e criação da classe *Cidade*; começo do design da página inicial usando o *RelativeLayout*.
- 05/11 às 11h00min: adaptação das classes *Grafo* e *Vertice*. Nesta, trocamos a string rótulo para ser um objeto genérico. Da mesma forma, fizemos a classe *BucketHash* ser genérica. Assim, conseguimos ler o arquivo de cidades e inserir na tabela hash.
- 05/11 às 13h20min: a classe *Lista* passa a implementar a interface *IEnumerable* e define o método *GetEnumerator()*. Dessa forma, pode-se usar um *foreach* para percorrer a lista.
- 12/11 às 9h10min: começamos a fazer o canvas com a imagem da Península Ibérica.
- 13/11 às 8h35min: começamos a fazer o layout da adição de cidades.
- 25/11 e 26/11: transição do Xamarin para o Android Studio, adaptando as classes para Java.
- 28/11 às 11h00min: classe *Aresta*.
- 01/12 às 12h00min: adaptação da classe *Grafo*, para que tenha uma matriz de *Aresta* em vez de uma matriz de *int*; inserção de vértices no grafo na hora da leitura do arquivo.
- 01/12 às 14h00min: criação da classe *Caminho*, retornada no método *getCaminho()* da classe *Grafo*.
- 02/12 às 11h00min: método *desenhaCaminho()* da classe *Desenhadora*, além do *btnBuscar.onClick()*.
- 03/12 às 9h00min: fizemos testes no método *getCaminho()* da classe *Grafo*, consertando alguns erros. O método agora funcionou.
- 03/12 às 11h00min: adição da classe *PesoCidades*, que é o tipo da matriz. Antes o tipo era *Aresta*, resultando em objetos *Cidade* a mais no *Grafo*. A classe *Aresta* tem como atributo um objeto *PesoCidades*.
- 03/12 às 19h00min: exibição dos dados do caminho gerado em um *TableLayout*.

## Erros e dificuldades

- 05/11: por alguma razão, os atributos que seriam usados para posicionar os elementos no *RelativeLayout* não estão sendo reconhecidos. Provavelmente está faltando baixar algo no Visual Studio.

Resolução: atualizei o projeto, então, os atributos foram reconhecidos.

- 05/11: tivemos dificuldades em ler um arquivo texto por meio do Android. O arquivo não era achado no construtor do *StreamReader*. Resolução: usar a pasta *Assets*, que tem o propósito de guardar arquivos que serão usados pelo programa.
- 12/11: não conseguimos desenhar a imagem na tela por meio do *Canvas*. Tivemos que pesquisar na internet.
- 25/11: decidimos abandonar o Xamarin, devido aos múltiplos erros que enfrentamos e não achamos soluções. Consideramos a ferramenta muito limitada e mal feita. Tentamos neste dia converter tudo o que foi feito até então para o Java, no Android Studio.
- 26/11: A pasta *Assets* do Android não permitia a escrita de arquivos. Por isso, tivemos de usar os métodos *openFileInput* e *openFileOutput*. No entanto, os arquivos texto devem ser escritos uma vez no programa antes de serem lidos.

## Conclusão

Com o fim do projeto, tivemos uma evolução significativa quanto ao manuseio de um *Canvas* em aplicações mobile, entendendo melhor como desenhar objetos gráficos na tela. Também aumentamos nosso conhecimento em relação aos arquivos texto em Android, lendo e escrevendo no armazenamento interno.

Além disso, entendemos melhor como os algoritmos de estruturas de dados funcionam, já que foi necessária uma adaptação dos mesmos conforme a proposta de caminhos entre cidades.

O projeto foi uma ótima oportunidade para treinar nossos conhecimentos.