

Ana Isabel Sampaio (AIS)

ESINF REPORT



**November 2022**

**2DN**

1210817, João Rodrigues

1210821, Mateus Fernandes

1180727, Ruben Martins

1200882, Gonçalo Teixeira

**INDÍCE**

[1. Diagrama de Classes 1](#_Toc116303297)

[2. Funcionalidades 2](#_Toc116303298)

[**2.1. Exercise 1** 2](#_Toc116303299)

[**2.2. Exercise 2** 2](#_Toc116303300)

[**2.3. Exercise 3** 2](#_Toc116303301)

[**2.4. Exercise 4** 4](#_Toc116303302)

[**2.5. Exercise 5** 6](#_Toc116303303)

[3. Possíveis Melhoramentos 8](#_Toc116303304)

**Índice de figuras**

[Figura 1- Método minCountries 3](#_Toc116303152)

# **Diagrama de Classes**

# **Funcionalidades**

## **2.1. Exercise 1**

Uma imagem com texto

Descrição gerada automaticamente

Uma imagem com texto

Descrição gerada automaticamente

## **2.2. Exercise 2**

O exercício 2 pretendia que para um dado fruto e dada uma x quantidade desse fruto devolver uma lista de países posteriormente ordenada por dois critérios. Como tal, o fruto e a quantidade são passados por parâmetros bem como o mapa por nos contruído com toda a informação proveniente do Excel.

Portanto, comecei por criar um sortedMap para ser retornado, mais um hashMap com a informação a recolher dos frutos e quantidade por cada ano de um dado país. Seguidamente comecei a minha pesquisa, comecei a percorrer os países, com um set de anos a guardar depois de encontrar os anos que pretendia, depois percorrer os anos e guardar a informação relativa aso frutos e quantidades no mapa a cima mencionado. De seguida percorri o item no mapa já com a informação toda e comparei a ver se ele era igual ao fruto passado por parâmetro. Se esta condição fosse verdadeira eu guardava numa variável temporária a quantidade desse fruto nesse mesmo ano, se fosse falsa ele voltava ao ciclo. Como ultima verificação e a pedida no enunciado verifiquei se essa quantidade era superior ou igual á quantidade passada por parâmetros.

Quando saía do ciclo for de anos o mapa a retornar guardava o pais em questão e o set de anos com as quantidades já verificadas >= á quantidade pedida.

Uma imagem com texto

Descrição gerada automaticamente

## **2.3. Exercise 3**

A funcionalidade referente ao exercício 3 permite-nos descobrir o número de países que, em conjunto consegue ter uma quantidade superior a uma certa quantidade fornecida pelo utilizador. Para tal, criamos o método minCountries, que recebe como parâmetros um long quantidade e um TreeMap com String como Key e um TreeSet de Year como Value, e retorna um int.

Começamos por criar uma variável int count, que será o nosso Contador a ser retornado, uma variável long minQuant, variável essa que será posteriormente utilizada para gerir os cálculos das quantidades dos países. Foi criado também um mapa que irá guardar os nomes dos países como Key e a quantidade total de produção de cada país como Value.

Após a criação das variáveis necessárias para a execução do método, foi necessário percorrer o mapa com os dados carregados do ficheiro CSV. Para isso, foram utilizados dois For Each, um a percorrer cada entry do mapa e outro a percorrer cada Set de years. Dentro do primeiro ciclo for os nomes dos países e o total da quantidade para cada país, são adicionados no mapa criado anteriormente, o totalValue. Logo de seguida, esse mesmo mapa foi ordenado por ordem descendente dos seus valores de quantidade.

Após termos o nosso mapa com os valores pretendidos, basta apenas percorrer o mapa e verificar o número mínimo de de países a ser devolvido pela função. Percorremos o nosso mapa sorted e verificamos se o 1º valor é maior que a quantidade, se sim o nosso count fica igual a 1 e esse valor e devolvido. Caso isso não aconteça, o valor é adicionado á variável minQuant criada anteriormente e o valor do count incrementado. Este ciclo é repetido até o valor do minQuant ser maior que a quantidade presente como parâmetro no método.

Uma imagem com texto

Descrição gerada automaticamente

**Figura 1- Método minCountries**

## **2.4. Exercise 4**

O exercício 4 pede-nos para criar uma funcionalidade que ao receber certas coordenadas geográficas, um Item, um Element e um Year, devolve todos detalhes da Area geograficamente mais próxima das coordenadas passadas como parâmetro.

Para tal, foi criado método nearestCountry. Numa primeira instância foi criada uma 2D Tree onde iremos armazenar as Areas com a os detalhes acima referidos. Em seguida, iremos percorrer a AVL de Areas, que nos foi passada por parâmetro, usando o método inOrder da mesma classe. Em seguida percorremos a AVL de Items presente nessa mesma Area e usamos uma condição para verificar se o Item passado como parametro existe na AVL de Items presente nessa mesma Area. Este processo irá ser usado também na AVL de Elements presente em cada Item e na AVL de Years presente em cada Element. Se todas as correspondências estiverem corretas iremos adicionar á 2Dt Tree a Area correspondente e as suas coordenadas geográficas.

Após a nossa Tree estar completa apenas necessitamos de chamar o método previamente criado na classe KdTree, o método findNearestNeighbour que recebe um double latitude e um double longitude, efetua os cálculos necessários e devolve um node, neste caso uma Area pois a Kd Tree criada é do tipo Area.

## **2.5. Exercise 5**

Uma imagem com texto

Descrição gerada automaticamenteMétodo mais importante do exercício 5

Neste método começo por atribuir uma lista de anos ao treeMap que recebo por parâmetro, e posteriormente converto essa lista de anos numa de itens, invoco o método contador das frutas que estão a ser tratadas.

Começamos por correr todas as frutas em cada ano e através do contador de itens, faço as comparações entre anos para obter o máximo valor absoluto da diferença de produção, que é retornado numa String e é impresso no main.

Uma imagem com texto

Descrição gerada automaticamenteMétodo para contar o número de frutas que estão a ser tratadas

Neste método convertemos o TreeMap com a informação numa lista e iteramos ano a ano e adicionamos as frutas a uma lista e verifica se essa fruta já foi adicionada, se não adiciona. Retorna o tamanho da lista, que é o número de frutas.

# **Possíveis Melhoramentos**