



RELATÓRIO DE ESTRUTURAS DE INFORMAÇÃO

Projeto 1 – Países/Covid-19

Síntese

Ao longo do documento iremos demonstrar o modo como gerimos a informação que nos é fornecida através de um ficheiro de Excel, bem como a explicação dos nossos métodos para responder aos diferentes requerimentos que a aplicação necessitaria ter, tendo em conta a eficiência.

Realizador por:
Fábio Fernandes (1191430)
Bárbara Pinto (1191507)

Introdução

Na disciplina de Estruturas de Informação foi-nos proposto para primeiro projeto, desenvolver uma aplicação que permita gerir a informação de vários países relacionada com a pandemia COVID-19. Assim como, aplicar os conhecimentos em Java Collection Framework para responder eficientemente aos requisitos que nos foram colocados ao longo deste projeto.

Redigir um relatório para explicitar as classes definidas e a justificação para as mesmas em termos de eficiência.

1. Requisito nº1 – Carregar e guardar informação do ficheiro Excel

Para este primeiro procedimento nós decidimos armazenar a informação recolhida do ficheiro Excel que nos foi disponibilizado na plataforma moodle em 6 classes, sendo elas: World, Continent, Country, Death, Case, Test e Smoker.

A estrutura deste programa consiste em utilizar as classes referidas anteriormente de maneira mais eficiente possível, dispersando assim os vários elementos de informação tendo como linha de pensamento, obter a informação necessária ao longo das mesmas. Um pequeno exemplo, de maneira a verificar o número de casos positivos de um determinado país, teríamos de seguir este caminho: World -> Continent -> Country e por fim aceder à classe Case para recolher a informação pretendida.

Um passo para atingir uma organização eficiente recorremos a Maps para obter uma pesquisa mais rápida, usando como Key a data, dessa maneira sabendo o país e uma data específica obtemos instantaneamente o valor relevante, seja ele nº de casos, nº de mortes ou nº de testes.

Frisar que em relação ao print da informação para o utilizador, utilizamos TreeMaps para ordenar a informação eficiente, alavancando as forças da própria Java Collections Framework.

2. Requisito nº2 – Apresentar uma lista de países ordenados por ordem crescente do número de dias para atingir os 50 000 casos positivos de COVID-19

Com base na nossa linha de pensamento, a maneira mais eficiente para obter a lista de países ordenados, seria percorrer a lista de países que foram colocados ao longo dos 6 continentes.

Acedendo a lista de continentes contida na classe World, podemos entrar na classe Continent, habilitando-nos a obter a localização de todos os países bem como o seu iso code, assim, só necessitaríamos de aceder à classe Case, contendo um Treemap com a informação organizada por data sendo a Key e o número total de casos os Values.

Tendo a data onde a condição se verifica (atingir 50 000 casos), só necessitamos de contar os dias desde o primeiro registo até ao registo dos 50 000 casos.

3. Requisito nº3 – Apresentar o total de novos casos assim como novas mortes por continente/mês

Neste caso em específico necessitamos de uma pequena particularidade encontrar a data de início bem como a data final de cada mês, para possibilitar uma contagem eficiente de todos os novos casos por mês.

Em adição criamos dois métodos distintos que retornam um Map cada, que partem como Key o valor de cada mês, ou seja, janeiro = 1, e a soma de todos os novos casos bem como mortes desse mesmo país.

Sendo assim, este ponto tornou-se simples, percorrendo os todos países dos diversos continentes, conseguimos disponibilizar a informação, retirando de cada Map o respetivo valor, por ordem alfabética e cronológica, disponibilizando o mês e suas respetivas mortes e novos casos.

4. Requisito nº4 – Devolver para cada dia de um determinado mês e para um dado continente, os países por ordem decrescente de casos positivos.

Toda a organização previamente concebida facilitou a conquista deste requisito, isto porque todos os métodos bem como tratamento de informação já estavam delineados, no entanto é necessário estruturá-la da maneira pretendida e eficientemente, o que não facilitou nada este objetivo.

Primeiramente questionamos o utilizador qual o continente e o mês que pretende obter informação, de seguida, através do método eis o que acontece. Acedemos á classe World e procuramos pelo continente desejado, obtendo assim a lista de países contida nele. Posteriormente o mês introduzido permite-nos através da data obter os valores dos casos positivos para cada dia do mês. Sendo assim organizamos a informação num TreeMap em que a Key será os casos e os seus Values o país de maneira a mostrar por ordem crescente de dias, mas mantendo a ordem decrescente dos casos positivos.

5. Requisito nº5 – Devolver numa estrutura adequada, todos os países com mais de 70% de população de fumadores, ordenados por ordem decrescente do número de novas mortes.

Por último, mas não menos importante, para responder a esta etapa, necessitaríamos de receber a data inicial para obter o total de fumadores, fazendo uma pequena adição dos fumadores femininos e masculinos. Posto isto, estabelecemos para o país a restrição de 70.0 (valor em percentagem que nunca pode passar os 100 valores) para a condição necessária.

Para o print da informação recolhida, toda ela foi enquadrada com o uso de um Map onde a Key será a percentagem de fumadores e os Values o país e um TreeMap onde se retira o total de novas mortes para eficientemente ordenar por ordem decrescente, onde se pode tirar uma boa conclusão de que nem sempre há um maior número de mortes, onde existe uma maior percentagem de fumadores.

Conclusão

Com este trabalho podemos concluir que as ferramentas da Java Collection Framework realmente são boas para pôr em prática porque não só conseguem proporcionar uma melhor organização de toda a informação, como podem melhorar significativamente a eficiência do programa que estamos a construir.

Funções que por outros métodos iriam ser bem mais lentos, com a Java Collection Framework conseguem se tornar muito mais rápidos.