Licenciatura em Engenharia Informática



Relatório Trabalho Prático 2

Francisca Morais(1181122)

João Marques(1192221)

Paulo Couto(1200587)

Dezembro 2021

# US 201

- As a Port manager, I which to import ports from a text file and create a 2D-tree with port locations.

Nesta User Story, foi-nos pedido que como um Gestor de Portos, importássemos um ficheiro csv contendo todos os portos, e em seguida, os colocássemos numa 2d Tree. Para tal, criámos o método readPorts, que recebe uma string com o ficheiro csv, onde iremos percorrer o ficheiro linha a linha, colocando a informação em cada variável do Porto, criando assim um ArrayList de Portos.

Este método tem complexidade de **O(log n),** visto ter um ciclo while.

Text

Description automatically generated

Figura 1 - Class CsvReader, method readPorts

Em seguida, de forma a inserir o ArrayList de Ports, retornado pelo método readPorts, numa 2d Tree, utilizamos o método “insertPorts”. Este método tem uma complexidade de **O(n),** pois contém um ciclo for, no entanto, este método chama um outro método, chamado buildTree. Este método também tem complexidade de **O(n)** devido à utilização de recursividade. Assim, o método “insertPorts” teria uma complexidade de **O(n)** + **O(n)** que seria = **2 O(n),** no entanto, em notação **Big Oh**, não interessa o valor das constantes, pelo que ficamos então com o valor de **O(n).**

Graphical user interface, text, application

Description automatically generated  
Text

Description automatically generated

Figura 2 - Class KdTreePort, method insertPorts

Figura 3 - Class KdTre, method buildTree

A complexidade da US 201 é então :

**O(log n) + O(n) = O(n)**