

Relatório Projeto 3.4 AED 2020/2021 Versão 1.0

Nome: Edgar Filipe Ferreira Duarte

Nº Estudante: 2019216077

TP (inscrição): PL8 Login no Mooshak: AED 2019216077

Nº de horas de trabalho: 2 H Aulas Práticas de Laboratório: 2 H Fora de Sala de Aula: 0 H

(A Preencher pelo Docente) CLASSIFICAÇÃO:

Comentários:

Estrutura de Dados Principal usada em cada sub-projeto:

PROJ 3.1 Array

PROJ 3.2 AVL Tree

PROJ 3.3 Splay Tree

Estruturas de Dados usadas	Array	AVL Tree	Splay Tree
VANTAGENS GERAIS (max 3)	<ul style="list-style-type: none">• Implementação simples	<ul style="list-style-type: none">• Árvores de pesquisa equilibradas• Complexidade de inserção e consulta $O(\log(n))$	<ul style="list-style-type: none">• Elementos mais acedidos ficam mais perto da raiz, demorando menos tempo a serem consultados posteriormente• Complexidade de inserção e consulta $O(\log(n))$
DESVANTAGENS GERAIS (max 3)	<ul style="list-style-type: none">• Complexidade de inserção é $\log(n)$• Precisa-se de garantir que todos os níveis da árvore estão completos	<ul style="list-style-type: none">• Inserção tem um <i>overhead</i> grande com grande quantidade de inserções	<ul style="list-style-type: none">• Muitas rotações (overhead grande)• O tempo de execução pode não ser constante (facilmente se obtém <i>outliers</i>)
Justificação para a escolha no PROJ 3.1	Uma merkle tree pode ser representada por um array, sendo que esta abordagem muito mais simples de implementar. Ao obter-se o número de elementos de entrada, conseguimos saber a quantidade total de nodos existentes e podemos representar esses nodos como index do array. É também mais fácil o cálculo das hashcodes usando este método.		
Justificação para a escolha no PROJ 3.2	O enunciado referia um cenário em que havia mais consultas do que inserções e onde não existiam usuários mais comuns. Assim, precisa-se de uma árvore equilibrada com boa complexidade temporal na consulta. Para este efeito existiam as árvores AVL e as árvores VP. Entre estas, as AVL são melhores na consulta, sendo assim escolhidas.		
Justificação para a escolha no PROJ 3.3	O enunciado referia um cenário em que havia usuários que acedem mais frequentemente a aplicação. Assim, foi escolhida uma árvore Splay, visto que esta estrutura leva qualquer nodo recentemente inserido ou consultado para a raiz da árvore. Assim, os nós mais visitados ficam mais perto da raiz, demorando menos tempo a serem consultados no futuro.		