

Relatório Projeto 2 V1.1 AED 2023/2024

Nome: Miguel Curto Castela

PL (inscrição): PL8

Email:uc2022212972@student.uc.pt

Nº de estudante: 2022212972

	Semana 1	Semana 2	Semana 3	Semana 4	Semana 5
Árvore Binária	X	X			
Árvore Binária pesquisa		X	X		
Árvore AVL				X	
Árvore VP					X
Finalização Relatório					X

2. Recolha de Resultados (tabelas)

Tempos de inserção da árvore binária

Tempo (µs) / nº de elementos	10000	20000	30000	40000	50000	60000	70000	80000	90000	100000
Crescente (10% repetido)	2850310	11594300	27025400	49354600	76756900	112190000	156406000	206142000	262967000	330113000
Decrescente (10% repetido)	2900910	11577400	26781100	49,088300	74469200	114972000	153347000	205396000	263289000	327320000
Random (10% repetido)	2678320	10936800	24787800	44065600	68225400	98988500	141756000	177870000	226650000	290339000
Random (90% repetido)	549875	2271260	5154090	9185730	14450800	20454800	27839100	37045100	46863200	58927300

Tempos de inserção da árvore binária de pesquisa

Tempo (µs) / nº de elementos	10000	20000	30000	40000	50000	60000	70000	80000	90000	100000
Crescente (10% repetido)	168463	753892	1610900	2968760	4392270	6337850	8828520	11640900	14932400	18331100
Decrescente (10% repetido)	147173	605472	1344530	2681090	3824830	5869890	8264080	10370800	13363000	16425500
Random (10% repetido)	2301,4	5874,9	8336,5	12955,6	15161,5	20971,1	28207,9	31520,6	40062,6	39907
Random (90% repetido)	490,16	1194,69	2303,55	3189,35	3917,9	4926,31	5841,64	5999,33	7209,59	9959,73

Tempos de inserção da árvore AVL

Tempo (µs) / n° de elementos	10000	20000	30000	40000	50000	60000	70000	80000	90000	100000
Crescente (10% repetido)	3136,3	8204,89	12211,4	15261,2	18720,6	26628,5	31622	34496,8	35205	49799,2
Decrescente (10% repetido)	2723,1	6023,6	9111,7	15983,4	16016,3	23822,1	24945,1	30903	33525,8	36751,6
Random (10% repetido)	5147,4	10341,4	19567,4	29082,6	35967,1	40448,4	45486,7	50636,7	59952,2	64717,1
Random (90% repetido)	848,32	2752,68	3328,93	4754,75	5639,15	6733	8175,04	10246,4	11077,7	12476,1

Tempos de inserção da árvore vermelha e preta

Tempo (µs) / n° de elementos	10000	20000	30000	40000	50000	60000	70000	80000	90000	100000
Crescente (10% repetido)	3064,8	7420,9	11892,5	16657,2	22567	26739,2	33121,9	35209,8	44553,1	48254,5
Decrescente (10% repetido)	2966,1	6687,1	10461,9	19396	20546,7	29424,8	30357	38223,9	41157	46701,3
Random (10% repetido)	4618,7	10457,7	14422,1	22963,5	28879,8	34901,5	37468	50089,5	49380,7	56679
Random (90% repetido)	826,6	1857,23	2728,93	3315,28	4213,04	5179,65	6140,83	7535,92	8284,82	10540,7

Número de rotações da árvore AVL

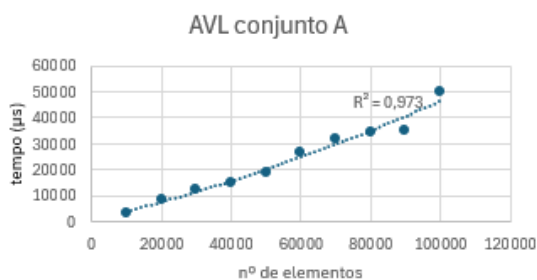
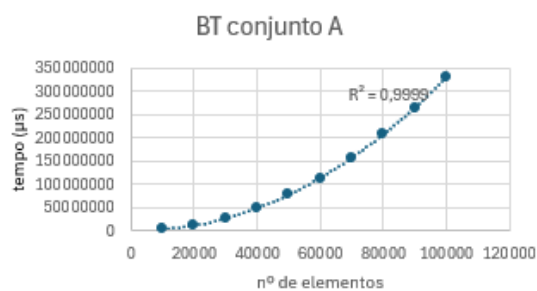
Rotações/n° de elementos	10000	20000	30000	40000	50000	60000	70000	80000	90000	100000
Crescente (10% repetido)	9986	19985	29985	39984	49984	59984	69983	79983	89983	99983
Decrescente (10% repetido)	9986	19985	29985	39984	49984	59984		79983	89983	99983
Random (10% repetido)	7095	13885	20603	27953	35077	42018	48805	56161	62813	69967
Random (90% repetido)	6979	13971	20890	27661	35033	41763	48879	55780	62520	69801

Numero de rotações da árvore vermelha e preta

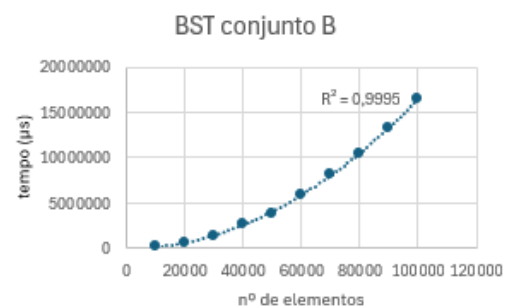
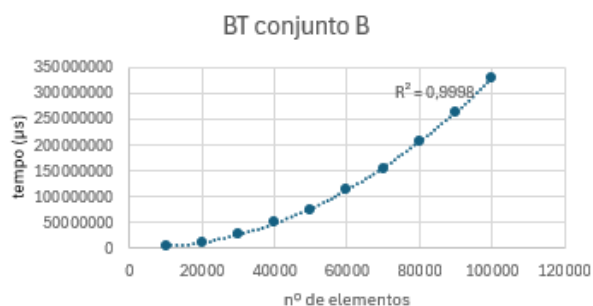
Rotações/nº de elementos	10000	20000	30000	40000	50000	60000	70000	80000	90000	100000
Crescente (10% repetido)	9976	19974	29973	39972	49971	59971	69970	79970	89970	99969
Decrescente (10% repetido)	9976	19974	29973	39	49971	59971		79970	89970	99969
Random (10% repetido)	5712	11775	17493	23180	28984	34851	40553	46592	52686	58168
Random (90% repetido)	5756	11588	17489	23167	29038	34626	40773	46772	52377	57916

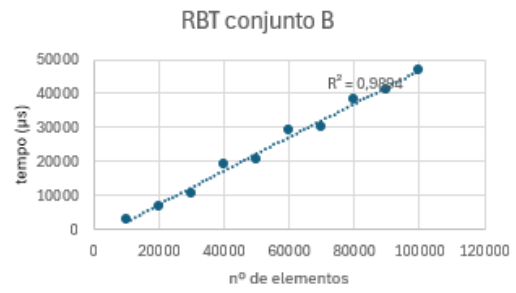
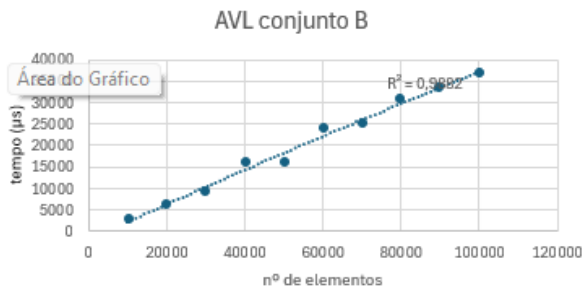
3. visualização de resultados (gráficos)

-Conjunto A

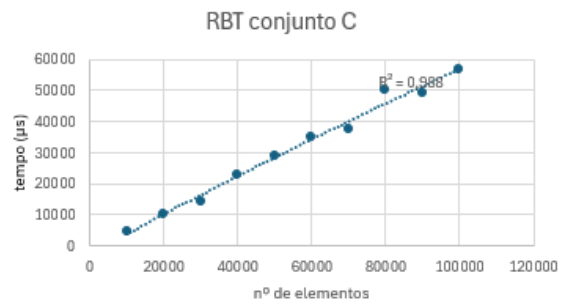
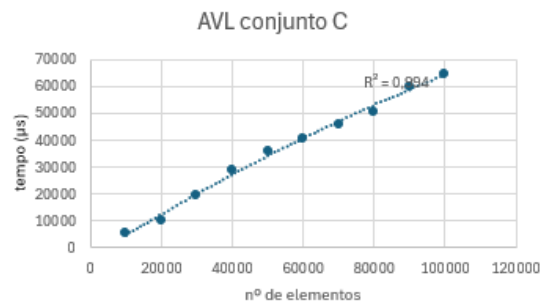
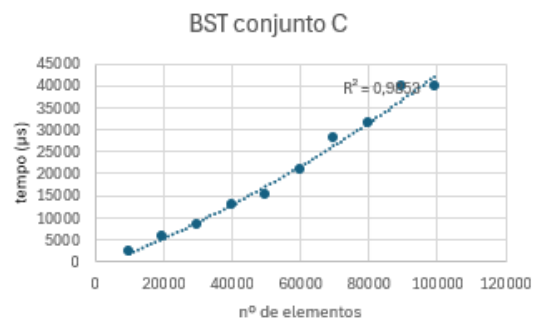
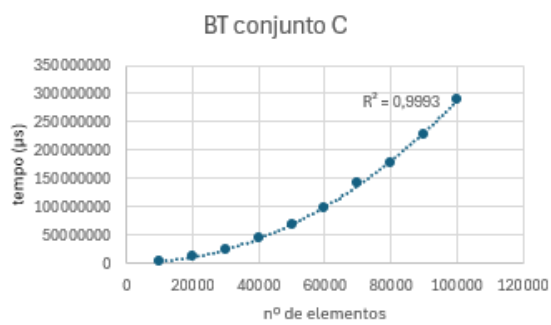


-Conjunto B

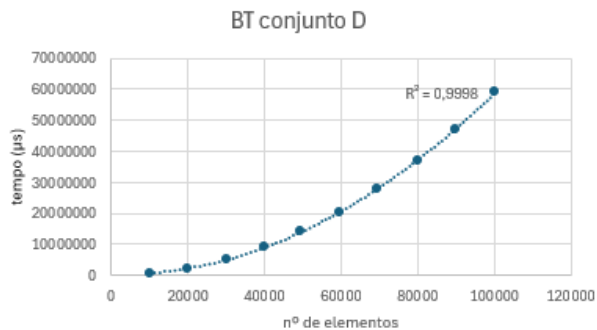


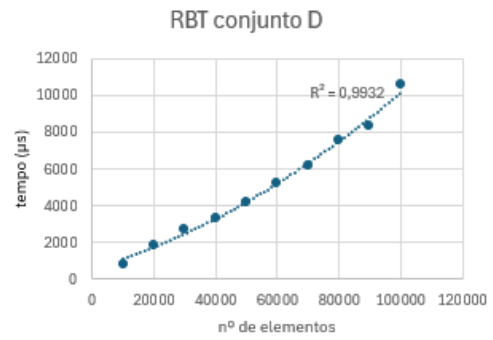
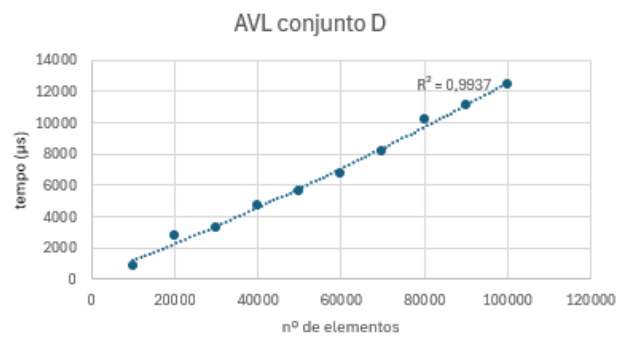


-Conjunto 3



-Conjunto 4





4.Conclusões

4.1 Tarefa 1

A complexidade de inserção na árvore binária, assumindo que já tem N elementos, é $O(N)$. Isto porque antes de inserir o elemento na árvore é necessário percorrer a árvore inteira, para verificar se o elemento já existe. Se o elemento existir o valor é retornado e não é inserido, mas se não existir é necessário percorrer a árvore novamente para inserir o elemento (no caso da binary tree com nó e filho com sci , de antes de ir para a direita ou esquerda, para ficar um nó + balanceada), cuja pior caso é, também, $O(N)$. Como tal a complexidade total de inserção de elementos é $O(N^2)$. Os melhores resultados foram obtidos para o conjunto D , uma vez que a verificação inicial é mais rápida. Os valores de N^2 comparam a análise teórica e é possível concluir que não é uma estrutura de dados boa para um grande quantidade de dados.

4.2 Tarefa 2

No pior caso, a inserção de um elemento na árvore binária de pesquisa é $O(N)$. Isto quando este tem o formato de uma lista (devido aos elementos inseridos de forma ordenada). No entanto, no caso médio, a complexidade é $O(\log N)$, pois a árvore é balanceada e é necessário "descer" $\log N$ níveis para encontrar o local de inserção do nó. Assim, a complexidade de inserção de N elementos é $O(N \log N)$. Para além disso, não é necessário fazer verificação inicial, pois caso o elemento já exista na árvore, vai ser encontrado e retornado durante a procura do local de inserção. Os resultados de N^2 estão de acordo com os resultados teóricos, e assim, conclui-se que para elementos ordenados escolher a BT e, por conseguinte, tem uma $O(N^2)$ performance.

4.3 Tarefa 3

A complexidade de inserção na árvore AVL é $O(\log N)$ uma vez que a árvore está sempre balanceada, e necessário descer $\log N$ níveis para chegar ao local de inserção, mas também é necessário subir o máximo máximo de níveis de modo a verificar a altura de balanceamento. Assim, a complexidade é $O(N \log N) + O(N \log N)$. Apesar de complexidade não muito alta, tem resultados piores do que a BT e BT , pois a BT é um pouco mais rápida e mantém a BT em ordem, mas não é. A verificação do fator de balanceamento $\{1, 0, -1\}$ obriga a travessia da

cria desde o mínimo até o máximo, aumentando o tempo de execução (mas melhorando os tempos de pesquisa). Os valores de M^2 e ressonância corroboram as previsões teóricas, sendo uma excelente estrutura para grandes conjuntos de dados, por exemplo.

4.4 Tarefa 4

A complexidade da inserção na árvore vermelha e preta é $O(\log N)$, uma vez que, a árvore é balanceada, mesmo que não seja de momento tão rígida como a AVL, tem no máximo de cerca de $\log(N)$ níveis até ao local de inserção. Assim a inserção de N elementos é de $O(N \log(N))$. Os resultados obtidos foram melhores dos que os de AVL uma vez que não sendo tão restrita, fez menos rotações, tendo um melhor tempo de execução (tanto um por de pesquisa). Os melhores resultados foram obtidos no conjunto D uma vez que vai ter menos elementos e ^{menos} níveis por descer. Os valores dos ressonância e M^2 corroboram com a previsão teórica. Foi a árvore com melhores performances.

5. Referencias

<https://www.youtube.com/watch?v=COZK7NATh4k>

<https://www.cs.usfca.edu/~galles/visualization>

<https://stackoverflow.com/>