

Alexandre Aparecido Scrocaro Junior
R.A.: 2135485

d)

Tabela 1

	N			
	1000	10000	100000	500000
ARN	0	0,008	0,097	0,496
ABB	0	0,3	0,45	1,263

Tempo de Execução (em s) para Ordenar Vetores com N Elementos em Ordem Crescente

Tabela 2

	N			
	1000	10000	100000	500000
ARN	8	12	15	17
ABB	999	9999		

Altura das Árvores Antes do Percurso Em-Ordem

e)

Tabela 3

	N			
	1000	10000	100000	500000
ARN	0,003	0,045	0,419	2,039
ABB	0,004	0,031	0,247	1,017

Tempo de Execução (em s, média +- desvio) para Ordenar Vetores Aleatórios com N Elementos

// A tabela 4 contém a mesma descrição que a tabela 2

f) Houve diferença significativa entre o tempo de execução entre os três métodos avaliados no caso da ordenação dos vetores inicialmente em ordem decrescente? Se sim, qual foi o melhor método? Qual foi o pior? Justifique suas respostas.

Sim, a ARN teve um tempo de execução muito menor, como demonstrado na primeira tabela. Dentre os dois métodos analisados, a ARN se mostrou a melhor e a ABB a pior, já que a ARN é uma árvore balanceada.

g) Houve diferença significativa entre o tempo de execução entre os três métodos avaliados no caso da ordenação dos vetores aleatórios? Se sim, qual foi o melhor método? Qual foi o pior? Justifique suas respostas.

Sim, dentre os dois métodos avaliados, o ABB teve um melhor desempenho, e o ARN teve um desempenho mais inferior, como visto na tabela 3. Pelo fato de se tratar de um vetor aleatório a árvore de busca binária conseguiu ter um melhor tempo médio de execução, já que, muito provavelmente, não caiu em casos degenerados.

h) Em cada um dos métodos (usando ABB, AVL, ARN), o tempo de execução foi muito diferente entre a ordenação dos vetores inicialmente em ordem decrescente e dos vetores aleatórios? Em qual método houve a maior variação? E a menor variação?

Sim, o ABB teve um tempo muito menor utilizando vetor aleatório, por ter casos sequenciais, o que aumenta o tamanho da árvore; e o ARN teve a menor variação, já que se trata de uma árvore equilibrada.