

Relatório RPEC Avaliação RA02

Daniela Lima

1 RESULTADOS:

Uso: java ArvoreAVL <inserir> <remove> <seed>

- Inserção de 100 elementos: 4 milissegundos

```
C:\Users\daniela\OneDrive\Desktop\RPEC>java ArvoreAVL 100 10 12345
Travessia pré-ordem da árvore construída:
51 28 16 6 2 1 0 3 10 8 9 12 15 21 18 17 19 25 26 32 29 31 43 41 39 42 49 44 50 80 71 55 53 52 54 64 58 56 67 66 70 74 7
3 76 75 77 89 84 82 81 83 87 88 92 90 91 97 96 94 99 98
Travessia pré-ordem após a exclusão do 10:
51 28 16 6 2 1 0 3 12 8 9 15 21 18 17 19 25 26 32 29 31 43 41 39 42 49 44 50 80 71 55 53 52 54 64 58 56 67 66 70 74 73 7
6 75 77 89 84 82 81 83 87 88 92 90 91 97 96 94 99 98

[Arvore com 100 elementos]
Tempo de inserção: 4 milissegundos
Tempo de remoção do elemento com chave 10: 0 milissegundos
```

- Inserção de 500 elementos: 1 milissegundo

```
Travessia pré-ordem da árvore construída:
241 108 44 17 8 3 1 2 6 7 10 9 14 12 16 27 21 19 22 25 37 30 28 29 32 33 39 38 43 80 59 54 52 48 46 51 53 56 55 57 58 74
70 63 60 72 71 73 77 76 75 79 94 84 82 81 83 87 86 85 88 91 99 97 96 98 104 100 103 105 158 132 117 114 110 109 111 115
116 121 119 118 120 126 125 128 145 137 134 133 136 141 138 140 144 143 151 148 147 155 154 152 156 157 187 175 170 166
164 160 165 169 167 173 172 174 178 177 176 184 182 186 215 207 191 189 188 190 198 195 203 201 205 210 208 213 214 228
224 219 216 223 225 227 232 231 229 238 237 236 239 389 328 274 258 250 249 246 253 251 252 255 254 257 265 263 261 270
267 273 302 287 281 279 275 284 283 286 294 291 290 292 300 298 296 299 301 318 312 305 303 306 314 314 313 316 321 320
319 325 323 326 364 341 336 332 331 329 333 335 339 338 337 340 351 349 347 344 348 350 357 354 353 355 356 361 358 360
363 375 370 366 365 368 367 369 371 374 380 378 376 379 384 382 381 383 388 386 461 428 403 397 392 390 396 395 399 398
400 402 412 409 406 404 411 423 416 414 418 417 425 424 426 442 434 431 430 432 439 438 435 440 452 449 444 443 447 450
454 453 458 456 459 483 474 466 463 462 465 468 467 470 472 479 477 476 481 480 482 490 486 484 488 494 492 491 497 495
499
Travessia pré-ordem após a exclusão do 10:
241 108 44 17 8 3 1 2 6 7 12 9 14 16 27 21 19 22 25 37 30 28 29 32 33 39 38 43 80 59 54 52 48 46 51 53 56 55 57 58 74 70
63 60 72 71 73 77 76 75 79 94 84 82 81 83 87 86 85 88 91 99 97 96 98 104 100 103 105 158 132 117 114 110 109 111 115 11
6 121 119 118 120 126 125 128 145 137 134 133 136 141 138 140 144 143 151 148 147 155 154 152 156 157 187 175 170 166 16
4 160 165 169 167 173 172 174 178 177 176 184 182 186 215 207 191 189 188 190 198 195 203 201 205 210 208 213 214 228 22
4 219 216 223 225 227 232 231 229 238 237 236 239 389 328 274 258 250 249 246 253 251 252 255 254 257 265 263 261 270 26
7 273 302 287 281 279 275 284 283 286 294 291 290 292 300 298 296 299 301 318 312 305 303 306 310 314 313 316 321 320 31
9 325 323 326 364 341 336 332 331 329 333 335 339 338 337 340 351 349 347 344 348 350 357 354 353 355 356 361 358 360 36
3 375 370 366 365 368 367 369 371 374 380 378 376 379 384 382 381 383 388 386 461 428 403 397 392 390 396 395 399 398 40
0 402 412 409 406 404 411 423 416 414 418 417 425 424 426 442 434 431 430 432 439 438 435 440 452 449 444 443 447 450 45
4 453 458 456 459 483 474 466 463 462 465 468 467 470 472 479 477 476 481 480 482 490 486 484 488 494 492 491 497 495 49
9
[Arvore com 500 elementos]
Tempo de inserção: 1 milissegundos
Tempo de remoção do elemento com chave 10: 0 milissegundos
```

- Inserção de 1000 elementos: 1 milissegundo

```
C:\Users\daniela\OneDrive\Desktop\RPEC>java ArvoreAVL 1000 10 12345
Travessia pré-ordem da árvore construída:
501 241 108 44 21 10 6 2 0 3 7 9 12 11 16 15 20 37 28 24 23 27 30 36 39 38 43 80 63 55 51 40 48 50 53 52 54 58 56 60 74 72 67 68 73 76 75 79 78 91 88 84 81 83 87 90 89 99 96 93 95 97 104 101 100 103 105 158 141 1
121 116 111 110 114 120 118 133 126 125 128 127 132 137 135 134 136 138 140 148 145 143 142 144 147 156 151 150 152 157 207 175 170 166 163 160 162 165 167 168 173 172 171 174 190 184 181 178 177 182 187 186 188
190 195 191 197 203 202 201 205 204 206 224 217 213 208 210 215 216 219 218 223 222 236 231 228 225 230 232 227 239 375 318 291 267 257 253 251 246 252 255 256 264 258 265 275 270 268 269 272 271 284 280 277
283 281 287 286 288 303 294 292 293 300 259 302 301 314 311 305 309 313 316 317 351 339 333 321 319 320 326 323 322 325 329 336 335 338 337 340 341 340 343 347 346 345 350 349 364 357 354 353 355 356 361 3
60 358 363 362 367 365 366 369 368 370 373 425 396 384 380 378 376 377 379 382 381 383 390 388 386 389 392 400 400 398 397 399 404 403 405 418 414 412 416 423 420 424 459 442 439 431 428 426 435 434 438 440 441
1 450 407 445 444 446 449 448 456 453 452 454 458 457 478 470 463 461 462 466 472 471 476 488 484 483 487 485 497 494 491 495 496 499 498 802 677 582 552 522 509 506 503 502 505 507 508 517 514 512 515 520 519
521 522 527 523 523 520 520 530 530 536 540 530 533 539 543 541 544 550 547 551 559 555 553 557 556 558 570 563 560 562 564 566 574 572 571 573 580 577 581 615 504 506 504 583 585 587 588 609 600 598 596 599 604
601 607 610 611 654 644 646 626 618 617 625 619 634 630 628 632 638 635 640 648 645 647 651 649 653 652 665 661 658 655 660 663 662 664 670 669 666 675 672 674 676 758 729 710 691 687 684 682 680 683 686 689 688 6
90 701 695 692 693 698 700 705 703 704 708 722 716 714 712 711 713 715 720 717 721 725 724 727 728 746 738 732 731 735 743 739 744 753 750 749 747 751 755 754 756 757 774 765 762 761 759 763 764 770 767 768 77
3 787 781 776 775 770 783 782 785 784 786 790 792 790 788 794 796 800 801 909 871 840 828 819 812 806 805 810 814 818 823 820 822 825 826 836 832 831 835 841 839 838 840 847 842 848 860 857 854 852 850 853 856
858 859 863 862 861 868 866 870 898 898 878 875 873 872 874 879 876 884 881 895 892 894 897 903 901 900 902 906 904 908 907 902 918 914 912 911 918 913 916 917 928 925 923 924 927 932 930 931 935 934 940 974 953
940 944 943 945 947 952 951 963 961 956 962 967 965 966 971 968 972 990 983 981 979 977 980 982 986 985 984 988 987 989 995 992 991 997 996
Travessia pré-ordem após a exclusão do 10:
501 241 108 44 21 11 6 2 0 3 7 9 16 12 15 20 37 28 24 23 27 30 36 39 38 43 80 63 55 51 40 48 50 53 52 54 58 56 60 74 72 67 68 73 76 75 79 78 91 88 84 81 83 87 90 89 99 96 93 95 97 104 101 100 103 105 158 141 1
121 116 111 110 114 120 118 133 126 125 128 127 132 137 135 134 136 138 140 148 145 143 142 144 147 156 151 150 152 157 207 175 170 166 163 160 162 165 167 168 173 172 171 174 190 184 181 178 177 182 187 186 188
190 195 191 197 203 202 201 205 204 206 224 217 213 208 210 215 216 219 218 223 222 236 231 228 225 230 232 227 239 375 318 291 267 257 253 251 246 252 255 256 264 258 265 275 270 268 269 272 271 284 280 277
283 281 287 286 288 303 294 292 293 300 259 302 301 314 311 305 309 313 316 317 351 339 333 321 319 320 326 323 322 325 329 336 335 338 337 340 341 340 343 347 346 345 350 349 364 357 354 353 355 356 361 360
358 363 362 367 365 366 369 368 370 373 425 396 384 380 378 376 377 379 382 381 383 390 388 386 389 392 400 400 398 397 399 404 403 405 418 414 412 416 423 420 424 459 442 439 431 428 426 435 434 438 440 441 4
50 407 445 444 446 449 448 456 453 452 454 458 457 478 470 463 461 462 466 472 471 476 488 484 483 487 485 497 494 491 495 496 499 498 802 677 582 552 522 509 506 503 502 505 507 508 517 514 512 515 520 519 521
522 527 523 523 520 520 530 530 536 540 530 533 539 543 541 544 550 547 551 559 555 553 557 556 558 570 563 560 562 564 566 574 572 571 573 580 577 581 615 504 506 504 583 585 587 588 609 600 598 596 599 604 601
607 610 611 654 644 646 626 618 617 625 619 634 630 628 632 638 635 640 648 645 647 651 649 653 652 665 661 658 655 660 663 662 664 670 669 666 675 672 674 676 758 729 710 691 687 684 682 680 683 686 689 688 690
701 695 692 693 698 700 705 703 704 708 722 716 714 712 711 713 715 720 717 721 725 724 727 728 746 738 732 731 735 743 739 744 753 750 749 747 751 755 754 756 757 774 765 762 761 759 763 764 770 767 768 773 7
87 781 776 775 770 783 782 785 784 786 790 792 790 788 794 796 800 801 909 871 840 828 819 812 806 805 810 814 818 823 820 822 825 826 836 832 831 835 841 839 838 840 847 842 848 860 857 854 852 850 853 856 858
859 863 862 861 868 866 870 898 898 878 875 873 872 874 879 876 884 881 895 892 894 897 903 901 900 902 906 904 908 907 902 918 914 912 911 918 913 916 917 928 925 923 924 927 932 930 931 935 934 940 974 953 940
944 943 945 947 952 951 963 961 956 962 967 965 966 971 968 972 990 983 981 979 977 980 982 986 985 984 988 987 989 995 992 991 997 996
[Arvore com 1000 elementos]
Tempo de inserção: 1 milissegundos
Tempo de remoção do elemento com chave 10: 0 milissegundos
```

- Inserção de 10000 elementos: 3 milissegundos

```
C:\Users\daniela\OneDrive\Desktop\RPEC>java ArvoreAVL 10000 10 12345
Travessia pré-ordem da árvore construída:
6251 3501 2004 806 384 173 79 43 28 14 6 3 2 0 4 5 10 8 11 13 22 15 16 26 27 39 32 31 29 37 33 38 41 40 55 47 46 45 51 50 49 52 66 62 58 57 59 63 65 71 67 78 72 133 106 93 88 84 81 80 82 86 85 90 89 92 91 97 9
4 95 100 99 101 103 112 110 108 107 109 111 123 117 113 116 121 119 122 128 126 130 158 142 136 134 135 138 137 139 140 151 145 144 149 155 153 152 154 157 156 167 164 159 163 165 166 171 170 172 207 188 1
70 177 176 178 170 184 182 181 186 185 187 209 202 196 195 190 198 197 200 204 213 212 211 214 204 235 232 219 218 221 220 232 223 230 234 240 243 237 236 240 246 245 248 252 251 253 262 250 250 255 257 261 25
```

```

5 9827 9819 9814 9813 9817 9815 9818 9823 9820 9822 9825 9832 9830 9829 9831 9833 9834 9840 9836 9838 9843 9841 9846 9845 9847 9879 9866 9857 9853 9850 9852 9855 9859 9858 9861 9875 9878 9868 9873 9872 9874 98
76 9878 9892 9886 9883 9882 9888 9884 9889 9887 9888 9891 9890 9898 9894 9893 9896 9895 9897 9908 9904 9902 9903 9905 9910 9909 9911 9963 9941 9920 9917 9914 9915 9910 9927 9923 9921 9925 9924 9926 9931 9929 9
939 9935 9932 9937 9949 9945 9942 9943 9948 9946 9958 9955 9950 9957 9961 9960 9962 9974 9968 9965 9964 9966 9967 9971 9969 9972 9973 9992 9980 9978 9976 9975 9977 9979 9986 9981 9984 9990 9989 9991 9996 9995
9993 9998
Travessia pré-ordem após a exclusão do 10:
6251 3501 2084 806 384 173 79 43 28 14 6 3 2 0 4 5 11 8 13 22 15 16 26 27 39 32 31 29 37 33 38 41 40 55 47 46 45 51 58 49 52 66 62 58 57 59 63 65 71 67 78 72 133 106 93 88 84 81 80 82 86 85 98 89 92 91 97 94 9
5 180 99 101 103 112 110 108 107 109 111 123 117 113 116 121 119 122 128 126 130 150 158 162 136 134 135 138 137 139 140 151 145 144 140 149 155 153 152 154 157 156 167 164 159 163 165 166 171 170 172 270 217 180 179
177 175 176 178 184 182 181 186 185 187 209 202 195 195 189 198 197 208 204 213 212 211 214 254 235 222 219 218 221 228 232 223 238 234 249 243 237 236 240 246 245 248 257 251 253 262 258 256 255 257 261 259 2
9878 9892 9886 9883 9882 9888 9884 9889 9887 9888 9891 9890 9900 9894 9893 9896 9895 9897 9908 9904 9902 9903 9905 9910 9909 9911 9963 9941 9920 9917 9914 9915 9910 9927 9923 9921 9925 9924 9926 9931 9929 9930
9935 9932 9937 9949 9945 9942 9943 9948 9946 9958 9955 9950 9957 9961 9960 9962 9974 9968 9965 9964 9966 9967 9971 9969 9972 9973 9992 9980 9978 9976 9975 9977 9979 9986 9981 9984 9990 9989 9991 9996 9995 999
3 9998
[Árvore com 10000 elementos]
Tempo de inserção: 3 milissegundos
Tempo de remoção do elemento com chave 10: 0 milissegundos

```

- Inserção de 20000 elementos: 7 milissegundos

```

C:\Users\daniela\OneDrive\Desktop\RPEC>java ArvoreAVL 20000 10 12345
Travessia pré-ordem da Árvore construída:
10025 5943 3389 2084 875 497 188 100 52 26 11 4 2 0 1 3 8 6 5 10 16 13 12 15 18 17 22 40 33 28 31 36 34 39 37 47 43 41 46 45 51 58 79 65 57 55 54 62 59 61 63 64 71 68 66 70 73 72 74 77 89 86 84 83 85 88 93 90
92 95 94 99 142 123 113 107 103 101 106 110 108 109 111 119 117 116 121 120 122 133 128 126 131 136 134 140 137 141 173 157 151 148 145 144 147 149 153 152 155 154 156 166 164 165 178 167 172 178 176 175 177 1
917 1915 1914 1916 1920 1919 1931 1924 1923 1922 1925 1935 1934 1938 1937 1940 1968 1949 1945 1943 1942 1944 1946 1948 1948 1964 1960 1955 1958 1962 1963 1965 1967 1981 1975 1971 1
969 19978 19972 19976 19977 19989 19986 19984 19982 19985 19987 19995 19992 19990 19993 19996 19998
Travessia pré-ordem após a exclusão do 10:
10025 5943 3389 2084 875 497 188 100 52 26 11 4 2 0 1 3 8 6 5 10 16 13 12 15 18 17 22 40 33 28 31 36 34 39 37 47 43 41 46 45 51 58 79 65 57 55 54 62 59 61 63 64 71 68 66 70 73 72 74 77 89 86 84 83 85 88 93 90 92
95 94 99 142 123 113 107 103 101 106 110 108 109 111 119 117 116 121 120 122 133 128 126 131 136 134 140 137 141 173 157 151 148 145 144 147 149 153 152 155 154 156 166 164 165 178 167 172 178 176 175 177 182
19915 19914 19916 19920 19919 19931 19924 19923 19922 19925 19935 19934 19938 19937 19940 19968 19949 19945 19943 19942 19944 19946 19948 19948 19964 19960 19955 19958 19962 19963 19965 19967 19981 19975 19971 1996
9 19978 19972 19976 19977 19989 19986 19984 19982 19985 19987 19995 19992 19990 19993 19996 19998
[Árvore com 20000 elementos]
Tempo de inserção: 7 milissegundos
Tempo de remoção do elemento com chave 10: 0 milissegundos

```

Remoção de determinado elemento: menos de 1 milissegundo

2 ANÁLISE DOS RESULTADOS:

Inserção: A inserção é eficiente para quantidades menores de elementos, como até 1000 elementos, onde o tempo é razoavelmente baixo. No entanto, o tempo aumenta substancialmente para 10000 e 20000 elementos. Isso é esperado, uma vez que a operação de inserção em uma Árvore AVL pode ser mais custosa à medida que a árvore cresce e requer rotações para manter o balanceamento.

Remoção: A remoção é muito eficiente, com tempo constante de menos de 1 milissegundo, independentemente do número de elementos. Isso ocorre porque a operação de remoção de um elemento de uma Árvore AVL envolve a mesma lógica de rotações usadas na inserção, e geralmente, é uma operação mais rápida.

3 ANÁLISE CRÍTICA DA IMPLEMENTAÇÃO:

A implementação da Árvore AVL é correta e segue os princípios da árvore de busca binária balanceada. No entanto, há alguns pontos que podem ser considerados:

Falta de balanceamento inicial: A implementação não realiza o balanceamento da árvore após a inserção de todos os elementos. Isso significa que a árvore pode não estar completamente balanceada após todas as inserções. Seria aconselhável adicionar uma função para reequilibrar a árvore após todas as inserções.

Cálculo de altura ineficiente: O cálculo da altura da árvore é feito recursivamente em cada nó. Isso pode ser otimizado mantendo o valor da altura em cada nó atualizado durante as operações de inserção e remoção, em vez de recalculá-lo a cada vez.

Inserção de elementos duplicados: A implementação permite a inserção de elementos duplicados, mas não fornece uma lógica para lidar com esses casos. Pode ser apropriado tratar elementos duplicados de forma específica, se desejado.

4 QUAL FUNCIONA MELHOR E EM QUAL SITUAÇÃO? POR QUÊ?

A Árvore AVL é uma estrutura eficiente para consultas, remoções e inserções em conjuntos de dados que precisam ser mantidos ordenados e permanecer balanceados. Ela é especialmente útil quando as operações de consulta são frequentes e você deseja manter uma árvore balanceada para evitar casos de pior cenário. No entanto, seu desempenho de inserção pode ser mais lento em comparação com outras estruturas, como a Árvore Binária de Busca não balanceada, em situações em que a inserção é a operação principal.

Portanto, a escolha entre a Árvore AVL e outras estruturas depende das operações que você planeja realizar com mais frequência e do tamanho dos conjuntos de dados. Se você precisa de um equilíbrio entre operações de inserção e consulta/remoção, a Árvore AVL é uma boa escolha. Se a maioria das operações é inserção e você não se preocupa tanto com o balanceamento, outras estruturas, como uma Árvore Binária de Busca padrão, podem ser mais eficientes.