

Exercício 1

Escrever o vetor que resulta imediatamente depois da sétima chamada para a função `merge()` quando usando o `mergesort` recursivo (TOP-DOWN) no seguinte vetor:

79 67 91 96 52 50 28 19 29 13 53 16

Exercício 2

Escrever o vetor que resulta imediatamente depois da sétima chamada para `merge()` quando usando o `mergesort` BOTTOM-UP no seguinte vetor:

81 16 13 22 91 99 18 53 83 71

Exercício 3

Quais das seguintes afirmações sobre o `mergesort` são verdadeiras? Justifique. A não ser especificado diferentemente, assuma que `mergesort` refere-se à versão recursiva TOP-DOWN do `mergesort` sem otimizações, usando a subrotina `merge` apresentada em aula.

- Um programador de sistemas embarcados pode preferir usar o `mergesort` BOTTOM-UP em vez de TOP-DOWN se está trabalhando em um ambiente em que recursão é muito caro
- `mergesort` BOTTOM-UP usa somente uma quantidade constante de espaço além do vetor de entrada
- No melhor caso, o número de comparações para o `mergesort` ordenar um vetor de N chave (não necessariamente distintos) é aproximadamente N .
- Considere quaisquer dois itens consecutivos no vetor que resulta após o uso do `mergesort` em um vetor com N chaves distintas. Então, aqueles dois itens foram comparados entre si em algum momento durante a ordenação.