

1 -

a) L <- conjunto S em ordem crescente

Enquanto (L <> 0)

Soma(L, menor, maior)

Se (soma == X)

Retorna 'SIM'

Se (soma > X)

Retira_Maior(L, maior)

Se (soma < X)

Retira_Menor(L, menor)

Fim Enquanto

Retorna 'NÃO'

2 -

a) - Para arranjos de 1 ou 2 elementos fornecidos como entrada, retorna o arranjo ordenado;

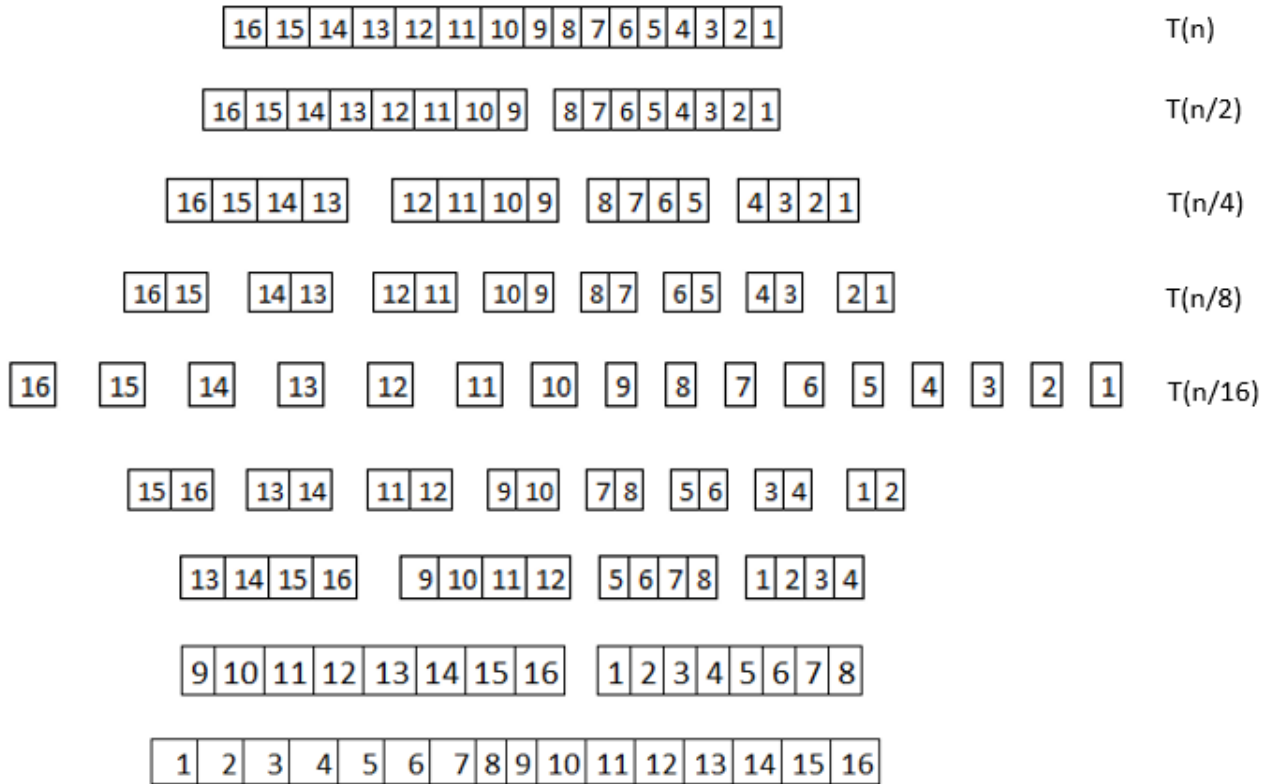
- Na primeira chamada recursiva (linha 6), ele coloca os menores elementos de α em A e os maiores elementos em B;

- Na segunda chamada recursiva (linha 7), ele coloca os menores elementos de β em B e os maiores em C, alterando os valores contidos em B;

- Após a terceira chamada recursiva (linha 8) o STOOGESORT terá reordenado os elementos de α , com isso todo o arranjo de entrada estará ordenado.

Portanto, o algoritmo ordena corretamente o arranjo de entrada.

3 -



A técnica de programação dinâmica não é capaz de acelerar o algoritmo porque apesar de ser um algoritmo recursivo e utilizar o método de divisão e conquista, ele utiliza a função de intercalação que possui tempo linear que é chamada repetidas vezes para realizar os merges, além de poder comparar números que já foram comparados.

4 -

a) Na primeira função, quanto maiores os valores de 'm' e 'n', maior será o número de execução da função, pois ela se divide em vários sub funções e geram somas sucessivas, portanto, sua complexidade é exponencial $O(2^n)$.

Na segunda função, há um ciclo 'para' dentro de outro ciclo 'para', gerando assim uma complexidade quadrática $O(n^2)$

b) A segunda função, mesmo sendo quadrática, acaba sendo mais eficiente do que a primeira função, pois ela possui maior desempenho com números muito grandes.