

MC458 - 1ª Prova Modelo 2 - 17/5/2013

- Complete a tabela abaixo com as complexidades dos algoritmos de ordenação mencionados.

Algoritmo	Pior caso	Melhor caso
BubbleSort		
InsertionSort		
HeapSort		
RadixSort		
CountingSort		

- Mostre que a complexidade de pior caso do QuickSort é $O(n^2)$, onde n é o número de elementos. Siga os seguintes passos:
 - Descreva o QuickSort; não é necessário descrever a fase de Partition em detalhe. Esboce somente.
 - Dê a complexidade de pior caso de cada trecho do QuickSort.
 - Junte os seus argumentos anteriores chegando à complexidade de $O(n^2)$.
- Analise a complexidade de pior caso do seguinte algoritmo recursivo de ordenação do vetor A com n elementos, descrito em pseudo-código. Como sempre, a chamada inicial é feita com $i = 1, j = n$.

```

SnailSort(A,i,j);
int aux,k;

if (i + 1) < j then {
    k= (j-i+1)/3;
    SnailSort (A, i, j-k);
    SnailSort (A, i+k, j);
    SnailSort (A, i, j-k)
}
else
    if A[i] > A[j] then {
        aux:=A[i]; A[i]:=A[j]; A[j]:=aux;
    }

```

- O algoritmo para encontrar o k -ésimo menor elemento de um vetor V descrito em sala divide inicialmente o vetor original em $\lfloor n/5 \rfloor$ vetores de tamanho 5, e prossegue com as demais tarefas. Se, nesse passo inicial, o vetor V fosse dividido em 5 vetores de tamanho $\lfloor n/5 \rfloor$ cada, isso alteraria a complexidade final do algoritmo?

5. Descreva a sua experiência no Lab1, quanto à implementação do algoritmo para encontrar o k -ésimo menor elemento de um vetor. Descreva principalmente as estruturas de dados usadas para montar os subvetores das chamadas recursivas subsequentes. Finalmente, diga qual a quantidade total de memória usada, como função de n ? Expresse essa quantidade assintoticamente caso você não tenha uma expressão exata.