

# AED2 2022 - ATIVIDADE PRÁTICA DE PROGRAMAÇÃO 03 - INSERTIONSORT RECURSIVO

Entrega: 10/05/2022 até 23:59:59

---

## Instruções:

1. E/S: tanto a entrada quanto a saída de dados devem ser "secas", ou seja, não devem apresentar frases explicativas. Siga o modelo fornecido e apenas complete as partes informadas (veja o exemplo abaixo).
2. Identificadores de variáveis: escolha nomes apropriados
3. Documentação: inclua cabeçalho, comentários e indentação no programa.
4. Submeta o programa no sistema judge utilizando acesso remoto via VPN: `http://judge.sjc.unifesp.br/aed2`, ou através de conexão direta: `http://kp.unifesp.br:9001/aed2/login`.
5. O código-fonte pode ser escrito em C, C++ ou Java.
6. Use alocação dinâmica, quando necessário, para alocar *arrays*. Memória alocada dinamicamente deve ser desalocada.
7. Nenhuma variável global deve ser utilizada.
8. **O código-fonte DEVE implementar a solução usando RECURSÃO. Soluções não recursivas não serão aceitas.**

## Descrição:

Deve-se implementar uma versão recursiva do algoritmos de ordenação em memória interna conhecido por *InsertionSort*, onde, a cada iteração/decida recursiva, um algoritmo de busca binária<sup>1</sup> deverá ser utilizado para localizar a posição exata onde o elemento chave deverá ser posicionado. A busca binária permite otimizar a tarefa de localizar a posição adequada da chave. A cada iteração recursiva o algoritmo deverá imprimir o nível de recursão alcançado, o valor da chave e a localização exata para o posicionamento do elemento chave. Para o critério de parada do algoritmo recursivo deve-se simplesmente retornar ao nível anterior em caso de lista vazia.

A relação de recorrência para o problema deve ser:

$$\begin{cases} T(n) = T(n - 1) + f(n) \\ T(0) = 0 \end{cases}$$

Onde  $f(n)$  representa o procedimento de busca binária e de deslocamento de elementos a cada iteração.

---

<sup>1</sup><https://pt.khanacademy.org/computing/computer-science/algorithms/binary-search/a/binary-search>

**Entrada:**

Em uma primeira linha deve-se ler o tamanho  $N$  da sequência de números a ser digitada, onde o tamanho máximo é  $N=1000$ .

Na próxima linha deve-se ler a sequência de inteiros separados por um espaço em branco, representando a sequência inicial  $P = (X1, X2, ..., XN)$  de  $N$  inteiros distintos.

**Saída:**

Os dados de saída consistem em  $M$  linhas, uma para cada nível de recursão alcançado, onde se deve exibir 3 dados separados por espaços: o nível de recursão, o valor da chave e a localização do elemento chave no vetor, conforme determinado pela busca binária. Após a exibição das linhas descritas acima, deve-se mostrar o vetor ordenado.

**Exemplos de entrada e saída:**

Exemplos de entrada	Exemplos de saída
5 1 5 3 4 2	4 5 1 3 3 1 2 4 2 1 2 1 1 2 3 4 5
6 3 5 2 1 4 6	5 5 1 4 2 0 3 1 0 2 4 3 1 6 5 1 2 3 4 5 6

Tabela 1: Exemplos de entrada e saída