

EP1 – Vale a pena ordenar? – ED FATEC SJC – Python
Versão diferente das passadas (2019-2)
(pode ser feito em dupla)

O algoritmo de busca binária é muito mais rápido que o de busca seqüencial, no entanto, para poder utilizá-lo precisamos antes ordenar os elementos do vetor, isto gasta certo tempo computacional. Quando vale a pena ordenar o vetor, para poder utilizar busca binária? Veremos na prática como medir isso verificando o número de buscas que são necessárias para que valha a pena ordenar o vetor. Isto é, fazer certo número de buscas sequenciais custa o mesmo tempo que fazer esse mesmo número de buscas binárias, mais uma ordenação anterior do vetor. Neste EP1 você deverá programar os algoritmos de busca binária, busca seqüencial e os algoritmos de ordenação por seleção, inserção, mergesort, quicksort e **sort nativo** do Python 3 e verificar esse número mínimo para cada método diferente de ordenação. Este EP1 deverá ser feito em Python 3.x e a saída deverá ser em formato texto (não utilizar interface GUI).

1. Diferente do EP1 da turma anterior você não precisa pedir opções ao usuário, basta mostrar todos os tempos na saída até passarem **30 segundos**.
2. Depois você iniciará um loop onde irá testando diferentes vetores aleatórios, começando com 5000 inteiros e incrementando de mais 5000 elementos a cada passo. Você deverá salvar uma cópia do vetor para auxiliar seus testes. Para cada algoritmo de ordenação você irá guardar o tempo consumido para a ordenação e depois rodar seguidamente busca seqüencial e busca binária até que o tempo de ordenação mais o tempo das buscas binárias igualem o tempo das buscas sequenciais.
3. **Aviso importante:** não é tolerado o plágio. A dupla que copiar ou emprestar o EP será reprovada. Caso você troque idéias com um colega, faça isso apenas verbalmente e de forma genérica. Certifique-se de construir o seu programa **desde o zero**, pois começar a codificar a partir de algum código pronto, exceto códigos vistos em aula, configura plágio. Alguns sites serão utilizados para verificação de plágio. O prof. possui as resoluções dos anos anteriores.
4. No seu programa não será necessário ler parâmetros, vocês poderão mostrar a saída abaixo no programa em Python, **incluindo o sort nativo, que não estão na saída abaixo**, mas devem ser incluídos no seu EP.

```
|-----[EP1 - Vale a pena ordenar?]-----|
| Algoritmo escolhido:      todos      Duracao dos testes:      30.00      |
| Aluno: Henrique F.P. Rosa - IME-USP      |
|      Tempos de Ordenacao      Numero de Buscas      |
|-----|-----|
|  n  | Insercao Selecao Merge. Quick. | Insercao Selecao Merge. Quick. |
|-----|-----|
| 5000 | 0.01  0.02  0.00  0.00      1745  5630  287  146 |
| 10000 | 0.03  0.05  0.00  0.00      3695  5419  452  36 |
| 15000 | 0.06  0.10  0.00  0.00      5220  8481  301  149 |
| 20000 | 0.12  0.18  0.01  0.00      6885  10767  340  29 |
| 25000 | 0.18  0.28  0.01  0.00      9041  14377  279  7 |
```