

Este é o CS50x

OpenCourseWare

Doar  (<https://cs50.harvard.edu/donate>)

David J. Malan (<https://cs.harvard.edu/malan/>)

malan@harvard.edu

 (<https://www.facebook.com/dmalan>)  (<https://github.com/dmalan>) 

(<https://www.instagram.com/davidjmalan/>)  (<https://www.linkedin.com/in/malan/>)

 (<https://orcid.org/0000-0001-5338-2522>) 

(<https://www.quora.com/profile/David-J-Malan>) 

(<https://www.reddit.com/user/davidjmalan>)  (<https://twitter.com/davidjmalan>)


Lab 3: Sort

You are welcome to collaborate with one or two classmates on this lab, though it is expected that every student in any such group contribute equally to the lab.

GitHub now requires that you use SSH or a personal access token instead of a password to log in, but you can still use `check50` and `submit50`! See cs50.ly/github (<https://cs50.ly/github>) for instructions if you haven't already!

Analise três programas de classificação para determinar quais algoritmos eles usam.

Quando fazer

No sábado, 1º de janeiro de 2022, 1h59 GMT-3  (<https://time.cs50.io/2021-12-31T23:59:00-05:00>).

Fundo

Lembre-se da aula em que vimos alguns algoritmos para classificar uma sequência de números: classificação por seleção, classificação por bolha e classificação por mesclagem.

- A classificação por seleção itera através das partes não classificadas de uma lista, selecionando o menor elemento a cada vez e movendo-o para seu local correto.
- A classificação por bolha compara pares de valores adjacentes, um de cada vez, e os troca se estiverem na ordem incorreta. Isso continua até que a lista seja classificada.
- A classificação de mesclagem divide recursivamente a lista em duas repetidamente e, em seguida, mescla as listas menores de volta em uma maior na ordem correta.

Começando

1. Faça login em ide.cs50.io (<https://ide.cs50.io/>) usando sua conta GitHub.
2. Na janela do seu terminal, execute `wget https://cdn.cs50.net/2020/fall/labs/3/lab3.zip` para baixar um arquivo Zip do código de distribuição do laboratório.
3. Na janela do seu terminal, execute `unzip lab3.zip` para descompactar (ou seja, descompactar) esse arquivo Zip.
4. Na janela do seu terminal, execute `cd lab3` para alterar os diretórios em seu `lab3` diretório.

Instruções

Fornecido a você três programas em C já compilados, `sort1`, `sort2`, e `sort3`. Cada um desses programas implementa um algoritmo de classificação diferente: classificação por seleção, classificação por bolha ou classificação por mesclagem (embora não necessariamente nessa ordem!). Sua tarefa é determinar qual algoritmo de classificação é usado por cada arquivo.

- `sort1`, `sort2` e `sort3` são arquivos binários, então você não será capaz de visualizar o código fonte C para cada um. Para avaliar qual classificação implementa qual algoritmo, execute as classificações em diferentes listas de valores.
- Vários `.txt` arquivos são fornecidos a você. Esses arquivos contêm `n` linhas de valores, revertidos, embaralhados ou classificados.
 - Por exemplo, `reversed10000.txt` contém 10.000 linhas de números que são invertidos `10000`, enquanto `random100000.txt` contém 100.000 linhas de números que estão em ordem aleatória.
- Para executar as classificações nos arquivos de texto, no terminal, execute `./[program_name] [text_file.txt]`.
 - Por exemplo, para classificar `reversed10000.txt` com `sort1`, run `./sort1 reversed10000.txt`.
- Você pode achar útil cronometrar seus tipos. Para fazer isso, corra `time ./[sort_file] [text_file.txt]`.

- Por exemplo, você pode correr `time ./sort1 reversed10000.txt` para correr `sort1` em 10.000 números invertidos. No final da saída do seu terminal, você pode olhar a `real` hora para ver quanto tempo realmente passou durante a execução do programa.
- Registre suas respostas `answers.txt`, junto com uma explicação para cada programa, preenchendo os espaços em branco marcados `TODO`.

Passo a passo



Dicas

- Os diferentes tipos de `.txt` arquivos podem ajudá-lo a determinar qual é o tipo. Considere o desempenho de cada algoritmo com uma lista já classificada. Que tal uma lista invertida? Ou lista embaralhada? Pode ajudar trabalhar com uma lista menor de cada tipo e percorrer cada processo de classificação.

► **Não sabe como resolver?**

Como verificar suas respostas

Execute o seguinte para avaliar a exatidão de suas respostas usando `check50`. Mas não deixe de preencher também suas explicações, que `check50` não serão marcadas aqui!

```
check50 cs50/labs/2021/x/sort
```

Como enviar

Execute o seguinte para enviar seu trabalho.

```
submit50 cs50/labs/2021/x/sort
```