

[Hora do Desafio]

[Desafios de Programação]

[Atividade avaliativa em squads, vale 1.0 ponto]



[o Desafio]

[DESAFIO]

Hash

Objetivo:

Criar uma base de dados de senhas ou nos e realizar operações de pesquisa e sobre a mesma.



Que tal iniciar por aqui:

Livro:

[Biblioteca Pearson] Estruturas de dados: algoritmos, análise da complexidade e implementações (Ascencio, A. F. G.; Araújo, G. S.)
- Capítulo 6 Tabelas Hashing

Artigo:

<https://pythonhelp.wordpress.com/tag/hash/>

Implementação:

https://panda.ime.usp.br/panda/static/pythonds_pt/05-OrdenacaoBusca/Hashing.html

Três squads:



Nível 1 - Squad Tô contigo

Python nativo com tratamento de colisão



Nível 2 - Squad Tô sacando

Lista criada com tratamento de colisão



Nível 3 - Squad Tô confiante

Lista encadeada com tratamento de colisão

Informar nomes dos componentes e

líder no [Escolha] Formação de times na sala virtual.

Desafio em
squads.

Usar a prática do
Pair
Programming
(adaptada).

[Explicação]

Prática
Pair Programming

do
eXtreme
Programming (XP)

Programação em pares (*Pair Programming*)

Duas pessoas implementam juntas o código diante do mesmo computador, revezando-se no teclado.

Uma pessoa é o "controlador", escreve o código, enquanto o outro, chamado de "observador" ("navegador"), analisa cada linha do código. Os papéis são revezados e, geralmente, tem-se um iniciante e um desenvolvedor mais experiente.

<https://youtu.be/a8WaP3Fwqa0>

Compartilhe o
ambiente de
programação!

1. **Planejarás a solução...** esboçarás e discutirás com o par!
2. **Consultarás o par e os livros** (referências na sala virtual).
3. **Realizarás vários casos de testes.**
4. **Adicionarás lindos comentários** ao código-fonte.
5. **Utilizarás boas-práticas** de programação!
6. **Serás original e criativo** desenvolvendo a própria solução.

Regras:



Para a **base de dados**: armazenar pelo menos 999 dados (aleatórios ou não).

Para a **pesquisa**: buscar por campo chave.

Para a **função hash**: é de escolha da squad.

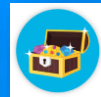
Para o tratamento de **colisão**: é de escolha da squad.

Para **resultado**: apresentar o tempo médio de busca (com e sem hash).

Observações:

Siga as regras... Elas foram definidas para que a squad desenvolva um **processo construtivo e evolutivo** de sucesso!

Regra de ouro:
Crie funções de autoria!



Casos de teste:



Documentar os casos de teste para apresentar em aula, alguns exemplos:

Caso de teste 1 - 0k

Entrada

Saída

Caso de teste 2 - 0k

Entrada

Saída

Caso de teste 3 - 0k

Entrada

Saída

Tempo médio

Pesquisa sequencial

Pesquisa hash

Observações:

Realizar **vários** casos de testes.

O que temos que entregar?

Solução pela sala virtual + **apresentação** em aula.



Na Sala virtual:

O **líder** deve postar a **solução conforme data indicada na sala virtual** (impreterivelmente).

Apresentar em aula no tempo de 5 minutos por time:

- a **demonstração** da execução, inicie apresentando os **casos de teste**;
- o restante do tempo, para **explicação do principal raciocínio do código-fonte** (função hash e tratamento de colisão).

Critérios de avaliação*:

(0,2) **Pensamento computacional** (atende o que pede o desafio, representa e manipula dados com hash de forma adequada, apresenta domínio do processo de construção da solução. **Requisito: utilizar Python e conhecimentos já apropriados até aqui.**

(0,2) **Correta execução do programa** (execução do programa sem erros, confiabilidade da resposta na execução de casos de teste).

(0,2) **Documentação do código-fonte** (identificação das funções e comentários nas principais linhas de comando - variáveis, regras e lógicas -, organização do código).

(0,2) **Originalidade** (autoria do código-fonte e capacidade de pensar na solução do problema).

(0,2) **Apresentação** (clareza e profundidade – detalhes - na apresentação).

* Todos os critérios acima estão condicionais ao cumprimento dos prazos (não serão aceitos desafios após o *deadline*).



Bom desafio!

[Não execute sem planejar antes!]



Principais referências para apoiar o desafio:

Sobre Python

- <https://www.python.org/>
- <https://docs.python.org/3/>
- [Livro] [Aprendendo python - Mark Lutz, David Ascher \(ver na sala virtual\)](#)