

Etapa de avaliação

1. Introdução

Pedimos para que, durante esta etapa, criem um repositório privado no GitHub com a solução do desafio descrito ao final do documento.

O repositório deve conter pelo menos:

1. O código principal
2. Um arquivo com testes
3. Um Readme com as instruções de execução

Note que a estrutura de arquivos é livre. Citamos 3 componentes principais que julgamos relevantes, mas a decisão de, talvez, modularizar e quebrar em mais arquivos é livre ao candidato.

A solução pode ser feita em Python 3+, Node.js (JavaScript) ou ANSI C (que deverá, caso escolhido, incluir um Makefile).

A escolha da linguagem não influencia a avaliação e não devem ser utilizadas bibliotecas que resolvem parte significativa do problema.

Os critérios são:

1. Funcionamento correto do programa (de acordo com especificações)
2. Qualidade do código:
 - a. Adequação do código a convenções da linguagem escolhida (PEP8 para o Python, por exemplo)
 - b. Legibilidade
 - c. Modularização
 - d. Reutilização
3. Cobertura de testes satisfatória

2. Desafio

Desenvolva uma pilha (estrutura de dados) de floats onde as operações de adicionar um item, remover do topo, consultar o item do topo e consultar o item de menor valor sejam **$O(1)$ em tempo**. **Não há limitação de espaço em memória** - mas otimizações em memória dão pontos extras.

Envie a resolução - ou qualquer dúvida sobre o problema - para jota@smartlike.com.br. O prazo é de **até 3 dias úteis**.