

Portas

Parte 1

Neste exercício, estudaremos o problema das *100 portas*. Temos 100 portas em sequência, numeradas 1-100, e todas começam trancadas. Agora passamos 100 vezes pelas portas. Na primeira passagem, você muda o estado das trancas (portas trancadas são abertas, portas abertas são trancadas) nas portas 1, 2, 3, 4, Na segunda passagem, você muda o estado das trancas nas portas 2, 4, 6, 8, . . . , e na terceira passagem, você muda o estado das trancas nas portas 3, 6, 9, 12, . . . , e assim por diante. Você faz isso até chegar a 100 (nessa última passagem apenas a última porta tem sua tranca modificada).

Matematicamente, isso significa que na m -ésima passagem, você troca o estado da tranca para cada m -ésima porta (quer dizer, as portas numeradas k de modo que $k \bmod m = 0$).

Na caixa abaixo, insira uma lista Python ordenada (menor para o maior) contendo os números das portas que estarão *desbloqueadas* após 100 passagens (nota: você pode querer escrever um programa Python para ajudar a responder essa pergunta!).

Portas abertas:

Parte 2

Você provavelmente notou um padrão interessante nos números acima! Nesta seção, investigaremos se esse padrão se mantém independentemente do número de portas consideradas.

Escreva uma função `nddoors` que recebe um único argumento inteiro representando n (o número de portas consideradas) e retorna uma lista de portas que estão desbloqueadas após n passagens pela porta. Por exemplo, se `1000` for passado como argumento, consideraremos o caso em que temos `1000` portas e fazemos `1000` passagens por elas.

Você deve resolver esse problema completamente, *não* apenas testar se o padrão acima se mantém ou não.

Antes de executar seu código, tente pensar se o padrão observado acima deve se manter conforme o número de portas muda.

Notas:

- Como na questão sobre "pi" do último conjunto, pode ser útil quebrar esse problema em partes menores. Uma estratégia seria primeiro simplesmente imprimir os números 1 a 100 (representando o valor m acima para cada passagem pelas portas). Daí, você pode seguir e, para cada valor de m , imprimir os números das portas cujas trancas serão alteradas naquela passagem. Quando tiver isso, vira uma questão de lembrar dos estados de cada porta.
- Você precisará de alguma estrutura para armazenar se as portas estão abertas e quais estão trancadas. Como você pode representar se uma porta está aberta ou fechada? Como você pode fazer isso para *todas as portas*?

Submissão

Quando estiver pronto, faça upload do seu arquivo Python no **Problema 2.3** no Gradescope. Lembre de nomear seu arquivo `p2_3.py`.

