

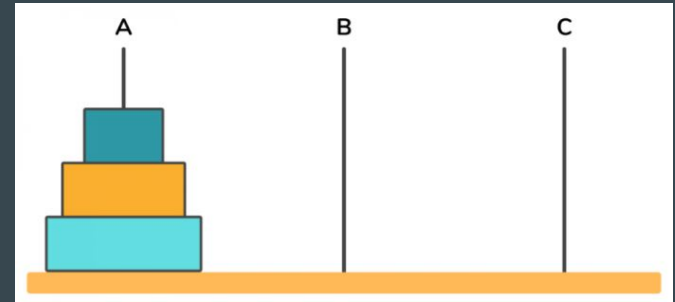
PERTANYAAN

...

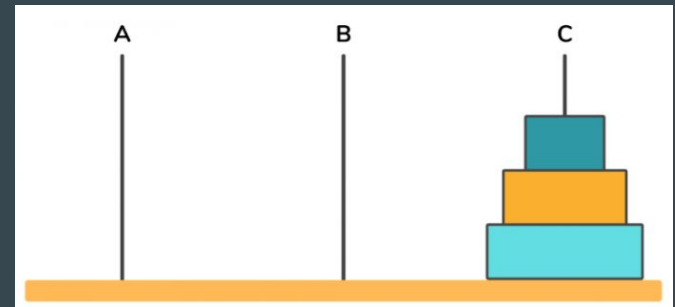
1. Pindahkan disk dari A ke C

- Hanya satu disk yang dapat dipindahkan pada satu waktu.
- Hanya disk paling atas yang dapat dipindahkan dari satu tumpukan ke atas tumpukan lain atau ke batang kosong.
- Disk yang lebih besar tidak dapat ditempatkan di atas disk yang lebih kecil.

STARTING POINT



EXPECTED RESULT

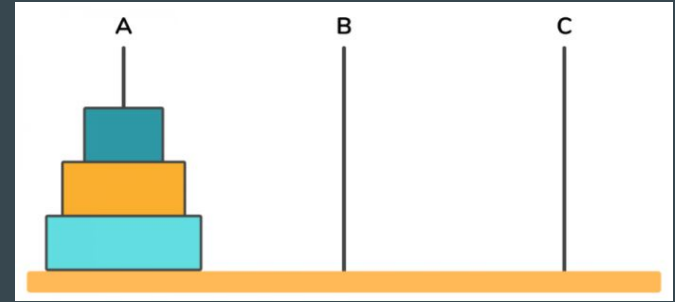


1. Pindahkan disk dari A ke C

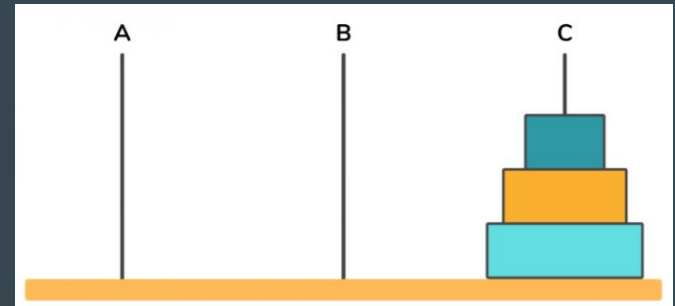
ILUSTRASI

- Pindahkan Disk 1 dari menara A ke menara B.
- Pindahkan Disk 2 dari menara A ke menara C.
- Pindahkan Disk 1 dari menara B ke menara C.
- Pindahkan Disk 3 dari menara A ke menara B.
- Pindahkan Disk 1 dari menara C ke menara A.
- Pindahkan Disk 2 dari menara C ke menara B.
- Pindahkan Disk 1 dari menara A ke menara B.

STARTING POINT



EXPECTED RESULT



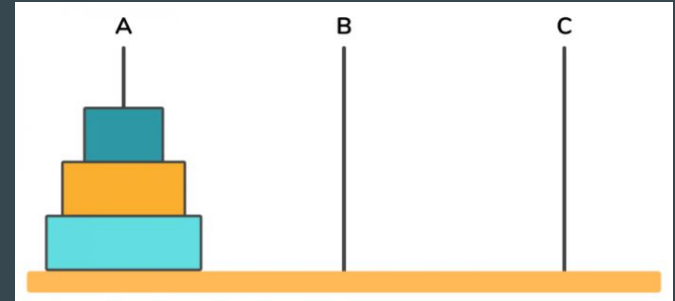
1. Pindahkan disk dari A ke C

// Return berapa minimal step yang diperlukan

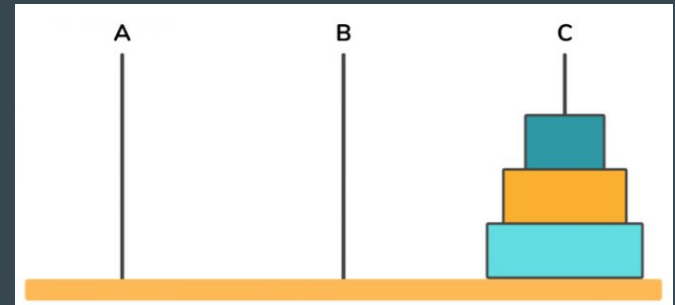
Function howManyStep(discs) {

}

STARTING POINT



EXPECTED RESULT



2. Optimize Reverse Geocoding

John telah membuat aplikasi absensi untuk karyawan harian menggunakan geolocation, dimana setiap karyawan yang membuka web aplikasi tersebut akan muncul alamat yang didapat dari reverse geocoding dari layanan Google Maps

<https://developers.google.com/maps/documentation/javascript/examples/geocoding-reverse?hl=id>

2. Optimize Reverse Geocoding

Setiap karyawan rata-rata per hari perlu absen di 10 lokasi berbeda.

Problem dari John adalah layanan Google Maps menerapkan harga \$5 per 1000 request lokasi.

$1 \text{ user} \times 30 \text{ hari} \times 10 \text{ lokasi} = 300 \text{ request}$

Kurang lebih biaya yang harus dikeluarkan John per 10 user adalah \$15. Dan biaya tersebut jauh diatas pendapatan subscription yang hanya \$10

Bagaimana cara anda untuk membantu John untuk menyelesaikan problem tersebut

2. Optimize Reverse Geocoding

Submit solusi kode anda dengan penjelasan mengapa anda menggunakan metode tersebut

3. Clean Architecture

- Buatlah code dengan 1 fitur CRUD (bebas) yang menerapkan Clean Architecture

<https://blog.cleancoder.com/uncle-bob/2012/08/13/the-clean-architecture.html>

