

## Lab Tutorial 3A

### Pinder

Nama berkas kode sumber	:	SDA<NPM>L3A_Jumat.java
Batas waktu eksekusi program	:	3 detik / kasus uji
Batas memori program	:	256 MB / kasus uji

Seorang Mahasiswa Fasilkom UUII bernama Riicky sedang kebingungan, karena orang tuanya memaksa dia menikah sedangkan dia belum memiliki pasangan. Oleh karena itu dia mencoba mencari pasangan hidupnya lewat aplikasi Pinder (Pasangan Idaman Coder).

Pada aplikasi Pinder, untuk mendapatkan kesempatan bertemu dengan pasangannya, maka seseorang harus menyelesaikan sebuah teka-teki. Teka-tekinya didefinisikan sebagai berikut:

1. Diberikan sebuah *String* yang terdiri dari 3 jenis karakter pembuka **L**, **V**, dan **<**, serta tiga jenis karakter penutup **O**, **V**, dan **3**
2. Didefinisikan juga **L** merupakan pasangan dari **O**. **V** merupakan pasangan dari **E**. Dan, **<** merupakan pasangan dari **3**
3. Sebuah pasangan karakter dikatakan **cocok** apabila *String* tersebut dimulai dengan jenis *String* pembuka dan ditutup dengan *String* jenis penutup pasangannya. Contoh: LO, VE, <3
4. Sebuah *String* dikatakan **tidak bagus** apabila terdapat bagian dimana *String* penutup pasangannya **tidak cocok**. Contoh: '<VLEO3' **tidak bagus** karena string **didalam** < dan 3 **polanya tidak cocok** (di dalam pasangan V dan E terdapat L, namun tidak ada O yang harusnya menutup L dan di dalam pasangan L dengan O terdapat E, namun tidak terdapat V yang harusnya menutup E), versi seimbang dari contoh tersebut salah satunya '<LOVE3' atau '<3LOVE'.
5. Jadi sebuah *String* dikatakan **bagus** apabila tidak terdapat pasangan karakter yang **tidak cocok** dan **isi dari sebuah pasangan *String* yang cocok** merupakan pasangan yang **cocok** juga.

Karena Riicky sangat ingin untuk bertemu dengan jodohnya lewat Pinder, bantulah Riicky untuk menyelesaikan teka-teki ini!

Anda tidak diperbolehkan menggunakan *class* yang ada pada Java Collection seperti List, Map, Set, Stack, Queue, dan lain-lain untuk menyelesaikan teka-teki ini. Penggunaan *class* tersebut berarti nilai 0.

### Format Masukan

Baris pertama dari input adalah nilai **N** yang menyatakan banyaknya calon pasangan potensial untuk Riicky.

**N** baris berikutnya merupakan string **S** yang terdiri dari karakter L, O, V, E, <, 3.

### Format Keluaran

Untuk masing-masing string, keluarkan 'BAGUS' apabila string tersebut merupakan string yang bagus, keluarkan 'TIDAK BAGUS' apabila string tersebut merupakan string yang tidak bagus.

### Constraint

- $1 \leq N \leq 10^3$
- $1 \leq |S| \leq 10^3$

### Contoh Masukan

```
3
<VLOE3
<VLE03
<<VVLL00EE33
```

### Contoh Keluaran

```
BAGUS
TIDAK BAGUS
BAGUS
```

## Penjelasan

1. Bagian terluar string < dan 3, cocok. Bagian dalamnya VLOE kita sebut X. Bagian terluar dari X yaitu V E, cocok. Bagian dalam dari X yaitu LO merupakan pasangan yang cocok juga sehingga keluarannya "BAGUS".
2. Bagian terluar string < dan 3, cocok. Bagian dalamnya VLEO kita sebut X. Bagian terluar X yaitu V dan O tidak cocok (atau bisa dilihat dari sudut pandang penjelasan soal diatas). Keluarannya "TIDAK BAGUS".

## Lab Tutorial 3B

### Bahagia

Nama berkas kode sumber : SDA<NPM>L3B\_Jumat.java  
Batas waktu eksekusi program : 3 detik / kasus uji  
Batas memori program : 256 MB / kasus uji

Setelah kalian membantu Riicky menemukan pasangannya lewat aplikasi Pinder. Sehabis kelas Riicky ingin menemui pasangannya dengan cara menuruni anak tangga disamping kelasnya. Akan tetapi Riicky penasaran berapa langkah tersingkat yang mungkin apabila setiap langkah memiliki ketentuan seperti berikut:

1. Riicky dapat menuruni 1 anak tangga dalam satu waktu. Contoh: Riicky dapat turun dari anak tangga ke-12 ke anak tangga ke-11.
2. Riicky dapat lompat dari anak tangga **N** ke anak tangga **B**, dengan syarat terdapat bilangan **A**, dimana  $A * B = N$ , dan  $\text{MAX}(A, B) = B$ . Contoh: Riicky dapat melompat turun ke anak tangga ke-6 atau ke-4 dari anak tangga ke-12 karena  $2 * 6 = 12$  ( $\text{max}(2, 6) = 6$ ) atau  $3 * 4 = 12$  ( $\text{max}(3, 4) = 4$ )

Riicky akan bertemu pasangannya pada anak tangga ke-0. Oleh karena itu bantulah Riicky bertemu pasangannya secepatnya!

### Format Masukan

Baris pertama dari input adalah **Q**, banyaknya kasus uji.

**Q** baris selanjutnya berisi bilangan **N**, yang artinya Riicky ingin turun dari tangga ke-**N** ke tangga ke-0.

### Format Keluaran

Jumlah langkah minimum untuk setiap masukan N.

### Constraint

- $1 \leq Q \leq 10^3$

- $1 \leq N \leq 10^4$

60% dari kasus uji

- $Q = 1$
- $1 \leq N \leq 500$

### Contoh Masukan

```
3
3
4
12
```

### Contoh Keluaran

```
3
3
4
```

### Penjelasan

1.  $3 \rightarrow 2 \rightarrow 1 \rightarrow 0$ . 3 langkah
2.  $4 \rightarrow 2 \rightarrow 1 \rightarrow 0$ . 3 langkah
3.  $12 \rightarrow 4 \rightarrow 2 \rightarrow 1 \rightarrow 0$ . 4 langkah

Berikut visualiasinya untuk  $N = 12$

### Hint:

#### 1. Agar Dapat Nilai

Dari tangga  $N$  apa saja pilihan kita? ( $N-1$  dan faktor-faktor dari  $N$ ).

#### 2. Agar Dapat 100

Jika kita sudah tahu jawaban untuk tangga ke  $N$  perlukah dihitung lagi? (Simpan jawaban di suatu struktur data).