



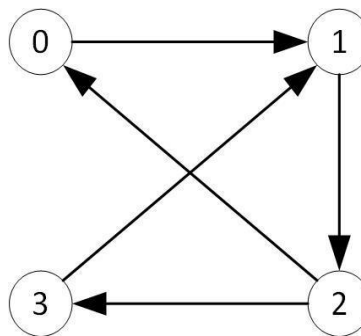
[C] Drone

Batas Waktu = 1 detik/*test-case*

Batas *Memory* = 64 MB

Deskripsi Masalah

Wishnu adalah seorang mahasiswa Fakultas Informatika di Universitas Telkom yang memiliki hobi yang cukup mahal, yaitu bermain *drone*. Dalam permainan yang dilakukannya, dia memakai N buah *drone* yang berkomunikasi secara otonom. Ini berarti Wishnu tidak perlu mengontrol pergerakan *drone-drone* tersebut menggunakan pengontrol jarak jauh. *Drone-drone* yang ada dilabeli dengan bilangan bulat i dengan $0 \leq i \leq N - 1$. Setiap *drone* i dan j untuk $i \neq j$ mungkin saja berkomunikasi secara langsung atau tidak langsung. Ketika sebuah *drone* i tidak dapat berkomunikasi dengan *drone* j secara langsung, maka mungkin kedua drone tersebut dapat berkomunikasi melalui beberapa *drone* perantara.



Gambar 1: Contoh diagram komunikasi yang terjadi antara empat drone

Komunikasi yang mungkin hanya berupa pengiriman pesan satu arah, yang berarti jika pesan dapat dikirim dari *drone* i ke *drone* j , maka hal sebaliknya belum tentu terjadi. Sebagai contoh, misalkan Wishnu memainkan empat *drone* yang dilabeli dengan 0, 1, 2, dan 3 (perhatikan ilustrasi pada Gambar 1). Pada Gambar 1 tersebut *drone* 0 hanya dapat mengirim pesan ke *drone* 1, *drone* 1 hanya dapat mengirim pesan ke *drone* 2, *drone* 2 dapat mengirim pesan ke *drone* 0 dan 3, serta *drone* 3 hanya dapat mengirim pesan ke *drone* 1. Wishu tidak ingin mengetahui apakah setiap *drone* i dapat mengirim pesan ke *drone* j , karena salah seorang temannya yang bernama Satrya



mengatakan bahwa masalah tersebut secara umum merupakan masalah NP-complete. Namun, ia ingin mengetahui apakah suatu *drone i* dapat mengirim pesan *drone j* dalam **tepat m** langkah komunikasi.

Sebagai contoh, kita melihat pada Gambar 1 bahwa *drone 0* dapat mengirim mengirim pesan ke *drone 2* dalam **tepat dua** langkah komunikasi. Pertama, *drone 0* mengirim pesan ke *drone 1*, kemudian *drone 1* mengirim pesan ke *drone 2*. Hal ini juga merupakan satu-satunya cara pengiriman pesan dari *drone 0* ke *drone 2* dalam tepat dua langkah. Contoh lainnya, *drone 1* dapat mengirim pesan ke *drone 2* dalam **tepat empat** langkah komunikasi. Ada dua acara pengiriman untuk kasus ini. Pertama dari *drone 1*, ke *drone 2*, ke *drone 0*, ke *drone 1*, lalu ke *drone 2* (kita tulis langkah ini sebagai $1 \rightarrow 2 \rightarrow 0 \rightarrow 1 \rightarrow 2$). Kedua dari *drone 1*, ke *drone 2*, ke *drone 3*, ke *drone 1*, lalu ke *drone 2*, atau $1 \rightarrow 2 \rightarrow 3 \rightarrow 1 \rightarrow 2$.

Wishnu hanya ingin mengetahui ada berapa banyak cara pengiriman pesan dari *drone i* ke *drone j* yang menggunakan **tepat m** langkah pengiriman pesan (tidak kurang dan tidak lebih dari *m*).

Format Masukan dan Keluaran

Masukan terdiri atas tiga bagian. Bagian pertama adalah sebuah baris yang memuat dua bilangan bulat *i* dan *j* yang dipisahkan dengan spasi dan memenuhi $0 \leq i, j \leq N - 1$, dengan *N* adalah banyaknya *drone* yang dimainkan Wishnu. Nilai *i* adalah label dari *drone* awal dan nilai *j* adalah label dari *drone* tujuan. Nilai *N* adalah bilangan bulat dengan $2 \leq N \leq 10$.

Bagian kedua merupakan sebuah bilangan bulat *m* ($1 \leq m \leq 10$), yang menyatakan panjang langkah pengiriman pesan dari *drone i* ke *drone j*.

Bagian ketiga dari masukan terdiri dari *N* baris yang setiap barisnya memuat *N* bilangan yang dipisahkan dengan spasi. Bilangan-bilangan ini bernilai 0 atau 1 yang mewakili mungkin-tidaknya mengirim pesan secara langsung (satu langkah) dari suatu *drone* ke *drone* lainnya. Secara spesifik, bilangan ke-*j* dari kiri pada baris ke-*i* bernilai 1 jika dan hanya jika kita dapat mengirim pesan secara langsung dari *drone* berlabel (*i* - 1) ke *drone* berlabel (*j* - 1).

Keluaran dari program adalah sebuah bilangan bulat yang menyatakan banyaknya cara mengirim pesan dari *drone* berlabel *i* ke *drone* berlabel *j* (yang dijelaskan pada baris pertama).



Contoh Masukan/Keluaran

Masukan	Keluaran
0 2 2 0 1 0 0 0 0 1 0 1 0 0 1 0 1 0 0	1
1 2 4 0 1 0 0 0 0 1 0 1 0 0 1 0 1 0 0	2

Penjelasan

Pada contoh masukan pertama *drone* awal adalah *drone* 0 dan *drone* tujuan adalah *drone* 2 (lihat Gambar 1). Kita ingin mencari banyaknya cara pengiriman pesan dari *drone* 0 ke *drone* 2 dalam tepat dua langkah pengiriman pesan. Baris ketiga sampai keenam menyatakan mungkin-tidaknya pengiriman pesan dari *drone* i ke *drone* j dilakukan dalam satu langkah komunikasi sebagaimana dijelaskan pada format masukan. Sebagai contoh, di sini kita memiliki bahwa dari *drone* 2 kita dapat mengirim pesan dalam satu langkah hanya ke *drone* 0 dan *drone* 3 saja. Dari informasi pada masukan, kita dapat mengetahui bahwa hanya ada satu cara untuk mengirim pesan dari *drone* 0 ke *drone* 2 dalam tepat dua langkah (lebih jauh, langkah yang dimaksud adalah $0 \rightarrow 1 \rightarrow 2$).

Pada contoh masukan kedua *drone* awal adalah *drone* 1 dan *drone* tujuan adalah *drone* 2 (lihat Gambar 1). Kita ingin mencari banyaknya cara pengiriman pesan dari *drone* 1 ke *drone* 2 dalam tepat empat langkah pengiriman pesan. Sebagaimana format masukan yang sebelumnya, baris ketiga sampai keenam menyatakan mungkin tidaknya pengiriman pesan dari *drone* i ke *drone* j dilakukan dalam satu langkah. Baris ketiga sampai keenam pada contoh masukan kedua ini sama persis dengan yang ada pada contoh masukan pertama karena kumpulan *drone* yang ditinjau sama (sebagaimana yang ada pada Gambar 1). Dari informasi pada masukan, kita memperoleh bahwa ada dua acara untuk mengirimkan pesan dari *drone* 1 ke *drone* 2, yaitu $1 \rightarrow 2 \rightarrow 0 \rightarrow 1 \rightarrow 2$ dan $1 \rightarrow 2 \rightarrow 3 \rightarrow 1 \rightarrow 2$.