



Мянган арал

Мянган арал бол Ява (Java) тэнгист байрладаг үзэсгэлэнт арлуудтай газар билээ. Уг газар 0-ээс $N - 1$ хүртэл дугаартай N аралтай.

0-ээс $M - 1$ дугаартай M тооны завийг ашиглан арлууд хооронд аялах боломжтой.

$0 \leq i \leq M - 1$ байх i дугаартай завийг $U[i]$ эсвэл $V[i]$ арлын аль нэгээс авч, $U[i]$ ба $V[i]$ арал хооронд аялахад ашиглаж болно. Тодруулбал, завь нь $U[i]$ аралд бэхлэгдсэн байгаа гэвэл түүнийг $U[i]$ арлаас $V[i]$ арал хүртэл ашигласны дараа $V[i]$ аралд бэхэлдэг. Үүний адилаар хэрэв завь $V[i]$ аралд бэхлэгдсэн бол түүнийг $V[i]$ -ээс $U[i]$ арал хүрэхэд ашигласны дараа $U[i]$ аралд бэхэлнэ. Анх завь $U[i]$ аралд бэхлэгдсэн байна. Ижил хос арлын хооронд хэд хэдэн завиар аялах боломжтой. Мөн нэг аралд хэд хэдэн завь бэхлэгдсэн байж болно.

Аюулгүй байдлын үүднээс завь нэг бүрийг ашигласны дараа засвар үйлчилгээ хийх шаардлагатай, тиймээс нэг завийг дараалан хоёр удаа ашиглахыг хориглодог. Өөрөөр хэлбэл, i дугаартай завийг ашигласны дараа уг (i дугаартай) завийг дахин ашиглахын өмнө өөр завийг ашиглах ёстой.

Би Dengklek зарим нэг арлуудаар аялахаар төлөвлөж байгаа. Дараах нөхцөл хангасан тохиолдолд түүний аялал **хүчинтэй** болно.

- Түүний аялал 0 дугаартай арлаас эхлэх ба бас энд (0 дугаартай арал) дуусдаг байна.
- Тэр 0 дугаартай арлаас өөр ядаж нэг арал дээр очсон байна.
- Аялал дууссаны дараа завь бүрийг аялалын өмнөхтэй адил аралд бэхэлсэн байна. Өөрөөр хэлбэл, $0 \leq i \leq M - 1$ байх аль ч i дугаартай завь нь $U[i]$ аралд бэхлэгдсэн байх ёстой.

Би Dengkle-д хамгийн ихдээ 2 000 000 удаа завиар аялах хүчинтэй аялалыг олж өгөх эсвэл тийм хүчинтэй аялал байхгүй эсэхийг тодорхойлоход тусална уу. Энэ даалгаварт заасан хязгаарлалтын дагуу (хязгаарлалт хэсгийг үзнэ үү) хэрэв хүчинтэй аялал байгаа бол 2 000 000-аас дээш удаа завиар аялахгүй байхаар хүчинтэй аялал бас бий гэдсэн баталгаа байгаа.

Хэрэгжүүлэлтийн мэдээлэл

Та дараах функцийг хэрэгжүүлэх ёстой:

```
union(bool, int[]) find_journey(int N, int M, int[] U, int[] V)
```

- N : арлын тоо.
- M : завины тоо.
- U, V : M урттай завинуудыг тодорхойлох массив.
- Энэ функц нь boolean эсвэл бүхэл тоон массив утга буцаана.
 - Хэрэв хүчинтэй аялал байхгүй бол уг функц нь false утга буцаана.
 - Хэрэв хүчинтэй аялал байгаа бол танд хоёр сонголт байна:
 - Бүрэн оноо авахын тулд функц нь хүчинтэй аяллыг илэрхийлэх хамгийн ихдээ 2 000 000 элементтэй бүхэл тоон массивыг буцаана. Тодруулбал, уг массивын элементүүд нь аялалд ашигласан завины дугаар байх ёстой (ашигласан дарааллаар).
 - Хэсэгчилсэн оноо авахын тулд процедур нь true, 2 000 000-аас дээш элементтэй бүхэл тооны массив эсвэл хүчинтэй аяллыг тайлбарлаагүй бүхэл тоон массивыг буцаадаг байна (Дэлгэрэнгүй мэдээллийг Дэд бодлого хэсгээс үзнэ үү.)
- Энэ функцийг яг нэг удаа дуудна.

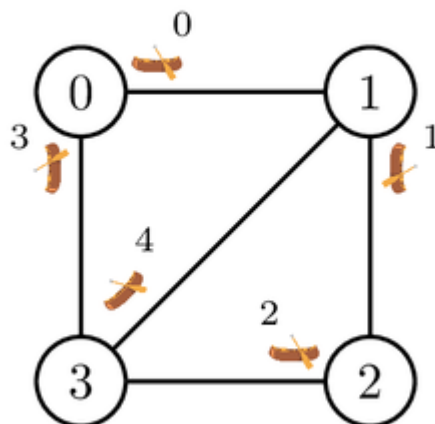
Жишээ

Жишээ 1

Дараах дуудалтыг авч үзье:

```
find_journey(4, 5, [0, 1, 2, 0, 3], [1, 2, 3, 3, 1])
```

Доорх зурагт арлууд болон завинуудыг харуулав.



Нэгэн боломжит хүчинтэй аялал нь дараах байдалтай байна. Ву Dengklek эхлээд 0, 1, 2 болон 4 гэсэн дарааллаар завиар аяласан. Үүний үр дүнд тэрээр 1 дугаартай арал дээр ирнэ. Үүний дараа Ву Dengklek нь 1 дугаартай аралд бэхлэгдсэн 0 дугаартай завиар дахин аялах боломжтой болно, учир нь тэрээр өмнө нь хамгийн сүүлд 0 дугаартай завийг ашиглаагүй. 0

дугаартай завиар аяласны дараа Bu Dengklek 0 дугаартай арал дээр байна. Гэсэн хэдий ч 1, 2 болон 4 завинуудыг аялалын өмнөх шиг арлууд дээр байрлуулаагүй. Тиймээс Bu Dengklek 3, 2, 1, 4 болон 3 завиар аялж аялалаа үргэлжлүүлнэ. Ингэснээр Bu Dengklek 0 арал дээр буцаж ирэх ба бүх завь аяллын өмнөхтэй ижил байдлаар аралд бэхлэгдсэн байна.

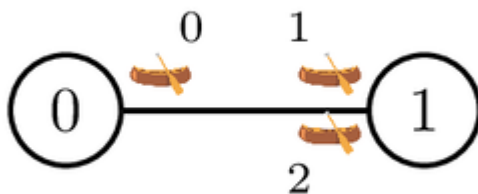
Тиймээс, функцийн буцаах утга $[0, 1, 2, 4, 0, 3, 2, 1, 4, 3]$ нь хүчинтэй аялалын утга болно.

Жишээ 2

Дараах дуудалтыг авч үзье:

```
find_journey(2, 3, [0, 1, 1], [1, 0, 0])
```

Доорх зурагт арлууд болон завинуудыг харуулав.



Bu Dengklek зөвхөн 0 дугаартай завийг ашиглаж, дараа нь 1 эсвэл 2-ын аль нэгийг ашиглах болно. Тэрээр 0-ын завиар хоёр удаа дараалан завиар явж чадахгүй гэдгийг анхаарна. Bu Dengklek 0 руу аль нэг завиар нь буцаж ирсэн гэе. Тэгвэл 0 аралд бэхлэгдсэн цорын ганц завийг Bu Dengklek дөнгөж сая ашигласан учраас шууд дахин ашиглаж чадахгүй юм. Хүчинтэй аялал байхгүй тул уг функц нь false гэсэн утга буцаана.

Хязгаарлалт

- $2 \leq N \leq 100\,000$
- $1 \leq M \leq 200\,000$
- $0 \leq U[i] \leq N - 1$ and $0 \leq V[i] \leq N - 1$ ($0 \leq i \leq M - 1$ байх i бүрийн хувьд)
- $U[i] \neq V[i]$ ($0 \leq i \leq M - 1$ айх i бүрийн хувьд)

Дэд бодлого

1. (5 оноо) $N = 2$
2. (5 оноо) $N \leq 400$. x ба y ($0 \leq x < y \leq N - 1$) гэсэн ялгаатай хос арал хооронд яг хоёр завиар аялах боломжтой. Тэдний нэг нь x арал дээр, нөгөө нь y арал дээр бэхлэгдсэн.

3. (21 оноо) $N \leq 1000$, M нь тэгш, $0 \leq i \leq M - 1$ байх **тэгш** i бүрийн хувьд i ба $i + 1$ дугаартай завийг хоёуланг нь $U[i]$ болон $V[i]$ арлуудын хооронд аялахад ашиглаж болно. i дугаартай завь нь $U[i]$ аралд, $i + 1$ дугаартай завь нь $V[i]$ аралд бэхлэгдсэн. $U[i] = V[i + 1]$ ба $V[i] = U[i + 1]$ гэсэн үг юм.
4. (24 оноо) $N \leq 1000$, M тэгш ба $0 \leq i \leq M - 1$ байх **тэгш** i бүрийн хувьд i ба $i + 1$ дугаартай завийг хоёуланг нь $U[i]$ болон $V[i]$ арлуудын хооронд аялахад ашиглаж болно. Эхэндээ хоёр завь нь $U[i]$ аралд бэхлэгдсэн байна. $U[i] = U[i + 1]$ ба $V[i] = V[i + 1]$ гэсэн үг юм.

Хүчинтэй аялал байгаа тестийн тохиолдол бүрийн хувьд таны шийдэл нь:

- хэрэв хүчинтэй аялал буцааж байвал бүтэн оноо авна,
- хэрэв true утга буцааж, 2 000 000-аас дээш тооны бүхэл тоон массив эсвэл хүчинтэй аяллыг тодорхойлоогүй массивыг буцаавал онооны 35%ийг авна.
- бусад үед 0 оноо авна.

Хүчинтэй аялал байхгүй тестийн тохиолдол бүрийн хувьд таны шийдэл нь:

- false гэж буцавал бүтэн оноо авна,
- бусад үед 0 оноо авна.

Дэд бодлого бүрийн эцсийн оноо нь дэд бодлого тестийн тохиолдлуудын хамгийн бага оноо гэдгийг анхаарна уу.

Жишээ grader

Жишээ grader нь оролтыг дараах форматаар уншина:

- мөр 1: $N M$
- мөр $2 + i$ ($0 \leq i \leq M - 1$): $U[i] V[i]$

Жишээ grader нь таны хариултыг дараах форматаар хэвлэнэ:

- Хэрэв find_journey нь bool утга буцаах бол:
 - мөр 1: 0
 - мөр 2: хэрэв find_journey нь false утга буцаасан бол 0, эсвэл бусад үед 1.
- Хэрэв find_journey нь int[]-ийг буцаасан бол массивын элементүүдийг $c[0], c[1], \dots, c[k - 1]$ гэж тэмдэглэнэ. Энэ үед жишээ grader-ийн хэвлэлт:
 - мөр 1: 1
 - мөр 2: k
 - мөр 3: $c[0] c[1] \dots c[k - 1]$