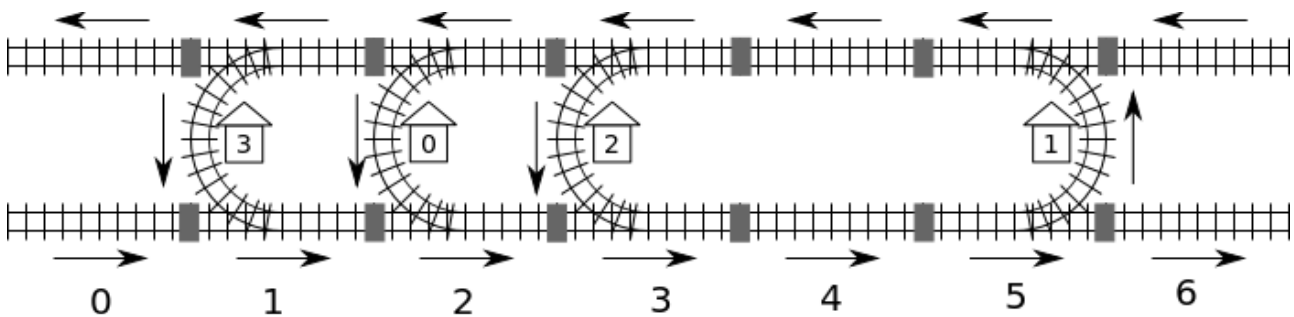




Төмөр зам

Тайвань нь арлын зүүн, баруун эргийг холбосон төмөр замын том шугамтай. Шугам m блокоос тогтоно. Үргэлжилсэн блокууд нь баруун захаасаа эхлэн $0, \dots, m - 1$ хооронд дугаарлагдсан. Блок бүр хойд талдаа баруун зүг чиглэсэн нэг чигийн замтай, урд талдаа зүүн зүг чиглэсэн нэг чигийн замтай ба тэдгээрийн хооронд галт тэрэгний буудал байж болно.

Гурван төрлийн блок байдаг. C төрлийн блок нь хойд талын замаасаа орж ирэн, урд талынхаа замруу гардаг галт тэрэгний буудалтай, D төрлийн блок нь урд талын замаасаа орж ирэн, хойд талынхаа замруу гардаг галт тэрэгний буудалтай, *empty* төрлийн блок нь ямар ч буудалгүй. Жишээ нь, доорх зурагт 0, 4, 6-р блокууд нь хоосон, 1, 2, 3-р блокууд нь C төрлийнх, блок 5 нь D төрлийнх байна. Хөрш блокууд нь *холбогч*-р хоорондоо холбогдох ба доорх зурагт будсан тэгш өнцөгтөөр харуулав.



Төмөр замын систем нь 0-с $n - 1$ хүртэл дугаарлагдсан n буудалтай. Бид ямар ч буудлаас ямар ч өөр буудалруу очиж чаддаг гэж үзье. Жишээ нь бид 0-р буудлаас 2-р буудалд 2-р блокоос эхлэн, дараагаар нь 3, 4-р блокийг урд талынх нь замаар өнгөрөн, тэгээд 5-р блокийг 1-р буудлаар өнгөрөн, дараагаар нь 4-р блокийг хойд замаар нь өнгөрөн, эцэст нь блок 3-т байрлах 2-р буудалд хүрнэ.

Тухайн буудалд хүрэх хэд хэдэн чиглэл байж болох тул нэг буудлаас нөгөөд очих замын урт нь *хамгийн бага* тооны холбогчийг замдаа дайрч өнгөрсөн чиглэлээр тодорхойлогдоно. Жишээ нь 0-р буудлаас 2-р буудалд хүрэх хамгийн богино чиглэл нь 2-3-4-5-4-3 блокуудаар дайрах бөгөөд 5 холбогчийг дайрч байгаа тул зай нь 5 байна.

Төмөр замын системийг компьютерын систем зохицуулдаг. Харамсалтай нь цахилгааны гэмтлийн дараагаар компьютер хаана буудлууд байсан болон тэдгээр нь ямар төрлийн блокуудад байсныг мартсан. Компьютерт байгаа цорын ганц баримт нь 0-р буудал байрлах блокийн дугаар ба тэр блок нь үргэлж C төрлийн блок байна. Аз болоход компьютер аль ч хотоос өөр аль нэг хот хүртэлх замын уртыг асууж болно. Жишээ нь, компютераас '0-р буудлаас 2-р буудал хүртэлх зай хэд вэ?' гэж асууж чадах ба тэрээр 5-г хүлээн авна.

Даалгавар

Та буудал бүрийн хувьд блокийн дугаар, түүний төрлийг нь тодорхойлох `findLocation` функцийг хэрэгжүүлэх ёстой.

- `findLocation(n, first, location, stype)`
 - `n`: Буудлын тоо.
 - `first`: 0-р буудлын блокийн дугаар.
 - `location`: n хэмжээтэй хүснэгт; та i -р буудлын блокийн дугаарыг `location[i]`-д хадгалах хэрэгтэй.
 - `stype`: n хэмжээтэй хүснэгт; та i -р буудлын блокийнх нь төрлийг `stype[i]`-д хадгалах хэрэгтэй: C төрөлд 1, D төрөлд 2.

Та буудлуудын төрөл, байрлалуудыг олоход хэрэг болох `getDistance` функцийг дуудаж болно.

- `getDistance(i, j)` нь i -р буудлаас j -р буудал хүрэх зайг буцаана. `getDistance(i, i)` нь 0 буцаана. `getDistance(i, j)` нь хэрэв i эсвэл j нь $0 \leq i, j \leq n - 1$ завсрын гадна байвал -1 буцаана.

Дэд даалгаврууд

Бүх дэд даалгавруудад блокийн тоо болох n нь 1,000,000-с хэтрэхгүй. Зарим дэд даалгавруудад `getDistance` функцийг дуудах тоо нь хязгаарлагдмал. Уг хязгаар нь дэд даалгавар бүрд өөр өөр. Хэрэв энэ хязгаарыг хэтрүүлбэл таны програм нь 'wrong answer' гэсэн хариулт авах болно.

ДЭД даалгавар	ОНОО	n	<code>getDistance</code> дуудах тоо	ТЭМДЭГЛЭЛ
1	8	$1 \leq n \leq 100$	хязгааргүй	0-с бусад бүх буудлууд нь D төрлийн блокт байрлана.
2	22	$1 \leq n \leq 100$	хязгааргүй	0-р блокийн баруун талд байрлах бүх буудлууд нь D төрлийн блокт байрлах ба 0-р блокийн зүүн талд байрлах бүх буудлууд нь C төрлийн блокт байна.
3	26	$1 \leq n \leq 5,000$	$n(n - 1)/2$	нэмэлт хязгаарлалтгүй
4	44	$1 \leq n \leq 5,000$	$3(n - 1)$	нэмэлт хязгаарлалтгүй

Хэрэгжүүлэх заавар

Та `rail.c`, `rail.cpp` эсвэл `rail.pas` гэж нэрлэгдэх яг ганц файл илгээх ёстой. Уг файл дээр дүрслэгдсэнийг доорх хэлбэрийн дагуу `findLocation` функцийг хэрэгжүүлнэ. Мөн та C/C++ хэрэгжүүлэлтэд `rail.h` толгой файлыг агуулах хэрэгтэй.

C/C++ програм

```
void findLocation(int n, int first, int location[], int stype[]);
```

Паскал program

```
procedure findLocation(n, first : longint; var location,  
stypе : array of longint);
```

getDistance функцийн хэлбэр дараах байдалтай.

C/C++ програм

```
int getDistance(int i, int j);
```

Паскал програм

```
function getDistance(i, j: longint): longint;
```

Жишээ шалгагч (grader)

Жишээ шалгагч нь доорх форматаар оролтыг уншдаг:

- мөр 1: Дэд даалгаврын дугаар
- мөр 2: n
- мөр $3 + i$, ($0 \leq i \leq n - 1$): $stypе[i]$ (C төрөлд 1 ба D төрөл 2), $location[i]$.

Жишээ шалгагч хэрэв findLocation функц дуусахад таны програмын тооцоолсон $location[0] \dots location[n-1]$ болон $stypе[0] \dots stypе[n-1]$ оролттой таарч байвал Correct гэж хэвлэнэ, хэрэв таарахгүй бол Incorrect гэж хэвлэнэ.