

D. Garden Decorations

Problem Name	Garden Decorations
Time Limit	7 seconds
Memory Limit	1 gigabyte

Дэтжи N байшинтай гудмаар өдөр бүр сургуульдаа ирж очдог. Байшин нь 0 -ээс $N - 1$ дугаартай. Одоогоор байшин i нь i хүнд хамаарна.

Үзэгдэх орчныг өөрчлөхийн тулд оршин суугчид өөр хоорондоо байшингаа солихоор шийджээ. i -р байшинд шилжихийн тулд a_i байшинд амьдарч байх ёстой.

Байшин бүр нь цэцэрлэгтээ шувууны хөшөөтэй. Хөшөө бүр нь 2 боломжит төлөвт байдаг: далавч нь дэлгээтэй эсвэл хумьсан.

Оршин суугч нар нь тэдний шувууны хөшөө ямар харагдаж байх талаар хүчтэй төсөөлөлтэй ба төсөөлөлтэй нь таарахгүй бол уг шинэ байшинд нүүж орохоос татгалздаг.

Дэтжи тэднийг нүүлгэх зорилгоор шувууны хөшөөнүүдийг зохион байгуулж туслахыг хүсэж байгаа болно.

Түүний тулд тэр дараах зүйлсийг хийнэ: Тэр гудмын дагуу явж байхдаа шувуудыг нэг нэгээр нь ажиглаад зарим хөшөөнүүдийг тохируулна (дэлгэсэн болон хумьсан байдлаар).

Гэртээ болон сургууль дээр завгүй байдаг тиймээс тэр өмнө нь явсан шувууны хөшөөний төлвийг санадаггүй.

Азаар тэр бичиж авсан a_0, a_1, \dots, a_{N-1} тиймээс аль оршин суугч хаашаагаа нүүхийг нь мэдэж байгаа.

Дэтжид оршин суугчдын төсөөлөлд хөшөөг нь тааруулахын тулд түүнд ямар шувуудыг тохируулах стратегийг хэлж өгч туслана уу.

Тэр гудамжаар хамгийн ихдээ 60 удаа явах боломжтой гэвч өндөр оноонд хүрэхийн тулд тэр цөөн удаа явах хэрэгтэй.

Гүйцэтгэл

Энэ бол олон удаа ажиллах бодлого, таны программ олон удаа ажиллана гэсэн үг.

Ажиллуулах бүрт та нэг мөрнөөс w ба N гэсэн хоёр бүхэл тоог уншина. Тэдгээр нь харгалзан явалтын индекс болон байшингийн тоо юм.

Таны программын эхний ажиллалтан дээр $w = 0$ байна. Хоёр дахь дээр $w = 1$ гэх мэтээр явна.

Оролтын хоёр дахь мөр дээр N тооны a_0, a_1, \dots, a_{N-1} байна. i -р байшингийн шилжих хүн a_i байшинд амьдарч байна. a_i -н нь 0-ээс $N - 1$ -ийн сэлгэмэл байна. Оршин суугч нь нүүхгүй байж боломжтой тул $a_i = i$ байж болно.

Оршин суугчид байшингаа нэг удаа л солино. Энэ нь бэхлэгдсэн тестийн тохиолдлын хувьд N утга болон a_i утгуудын жагсаалт нь бүх ажиллуулалтын хувьд ижил байна гэсэн үг юм.

Эхний ажиллуулалт

Таны программ эхний ажиллуулалтанд $w = 0$ байна. Энэ ажиллуулалтаас та нэг бүхэл W ($0 \leq W \leq 60$) тоо гаргана. Энэ нь Детжигийн байшингуудын урдуур хэдэн удаа явах нийт тоо байна. Үүний дараа таны программ гарах хэрэгтэй. Тэгэхээр таны программ W удаа ажиллана.

Дараалсан ажиллуулалт

Таны программын дараагийн ажиллагаан дээр $w = 1$ байна. Дараа нь $w = 2$ болно. Ингэж явсаар хамгийн сүүлийн ажиллагаан дээр $w = W$ болно.

w, N болон the a_0, a_1, \dots, a_{N-1} -г уншсаны дараа Детжи гудамжаар явж эхлэнэ.

w сондгой байвал, Детже гэрээсээ сургууль $0, 1, \dots, N - 1$ энэ дарааллаар явна.

Таны программ b_0 мөрийг уншвал 0 эсвэл 1 байх ба 0 дэх байшингийн хөшөөний идэвтэй төлөв байна. b_0 уншсаны дараа таны гаралт 0 эсвэл 1 байх ба энэ нь b_0 -ын шинэ утга юм.

Таны программ b_1 мөрийг уншвал байшин 1-ийн төлөв байх ба гаралт нь b_1 -ийн шинэ утга байна.

Энэ үйлдлийг N байшин тус бүр дээр хийнэ. Хамгийн сүүлийн байшингийн хажуугаар Детжи өнгөрсний дараа (b_{N-1} -г уншаад, бичээд) таны программ гарна.

Таны программ b_{i-1} утгыг бичээд дараа нь b_i утгыг унших ёстой.

w тэгш байвал, Детжи сургуулиасаа гэрлүүгээ явах ба байшингуудыг $N - 1, N - 2, \dots, 0$ гэсэн эсрэг дарааллаар явна.

Процесс w сондгой байхтай ижил байх ба b_{N-1} уншаад бичээд b_{N-2} гэх мэт цааш явсаар b_0 хүртэл явна.

$w = 1$ байх үед оролтын утгууд b_0, b_1, \dots, b_{N-1} шувууны хөшөөнийнүүд эхний төлөвт байна. $w > 1$ үед оролтын утгууд b_0, b_1, \dots, b_{N-1} нь таны программд таны өмнөх программ ажиллаад тохируулсан утгууд байх болно.

Төгсгөлд нь таны программ сүүлийн удаа ажиллаад дуусна. Бүх i -гийн хувьд b_i утга нь b_{a_i} -ийн анхны утгатай тэнцүү болсон байх ёстой. Тэгэхгүй тохиолдолд буруу хариулт авна.

Тайлбар

$W + 1$ удаа тусдаа таны программ ажиллахдаа нийлбэр хугацааны цагийн хязгаарлалтаас хэтэрвэл таны илгээлт хугацааны хязгаарлалт хэтэрсэн алдаа заана.

Мөр бүрийн хэвлэлтийн дараа стандарт гаралт дээр flush үйлдлийг хийхгүй бол таны программ хугацааны хязгаарлалт хэтэрсэн байдлаар дүгнэгдэнэ.

Пайтон хэлэнд `input()` уншина. C++ хэлэнд `cout << endl;` мөр нь шинэ мөр рүү шилжихдээ flush үйлдлийг давхар хийнэ. `printf` ашигласан үед `fflush(stdout)` мөрийг ашиглаарай.

Хязгаарлалт болон Оноо

- $2 \leq N \leq 500$.
- Хамгийн ихдээ $W \leq 60$ ашиглана.

Таны шийдэл тестийн бүлгүүдийн олонлогоор тестлэгдэнэ. Тэр бүртээ оноо авна. Тестийн бүлэг бүр нь тестийн тохиолдлуудын олонлогоос тогтоно. Тестийн бүлгээс оноо авахын тулд тестийн бүлгийн бүх тестийн тохиолдлуудыг шийдвэрлэх ёстой.

Бүлэг	Дээд оноо	Хязгаар
1	10	$N = 2$
2	24	$N \leq 15$
3	9	$a_i = N - 1 - i$
4	13	$a_i = (i + 1) \bmod N$
5	13	$a_i = (i - 1) \bmod N$
6	31	Нэмэлт хязгаарлалт байхгүй

Бүх тестийн бүлэг бүрд таны программ зөв шийдвэрлэвэл, та дараах оноог дараах томъёон дээр тулгуурлан авна:

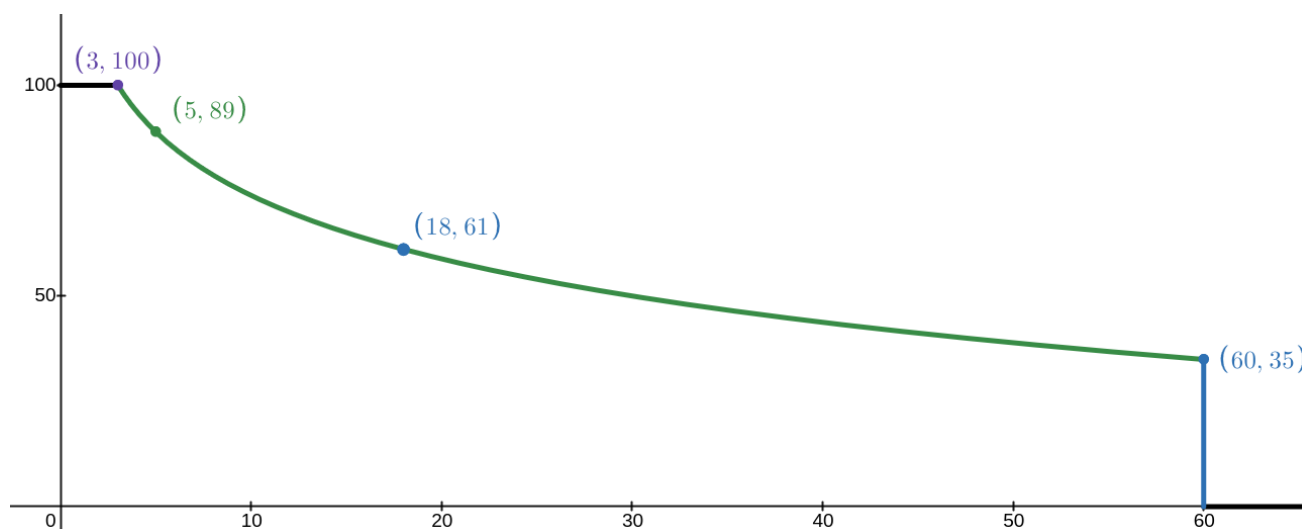
$$\text{score} = S_g \cdot \left(1 - \frac{1}{2} \log_{10}(\max(W_g, 3)/3)\right),$$

S_g тестийн бүлгийн дээд оноо, W_g бол тестийн бүлгийн тестийн тохиолдуулдын хувьд W -гийн хамгийн их утга.

Таны оноо тестийн бүлэг бүрийн хувьд хамгийн ойр бүхэл тоонд таарна.

Доор харуулсан график нь W -гийн утга нь ижил байх бүх тестийн бүлгийг шийдсэн үед таны программын авах онооны W -гаас хамаарах функцийг график юм.

Өөрөөр хэлбэл, энэ бодлого дээр 100 оноо авахын тулд та $W \leq 3$ байх тестийн тохиолдол бүрийг шийдэх ёстой.



Тестийн хэрэгсэл

Таны шийдлээ тестчлэх ажлыг хялбаршуулахын тулд бид энгийн хэрэгслийг татах боломжтойгоор байрлуулсан. Каттис бодлогын хуудасны доод хэсгээс "attachments" хэсгийг үз. Хэрэгсэл ашиглалт нь сонголтын байна. Каттис дээрх албан ёсны онооны программ тестийн хэрэгслээс өөр гэдгийг анхаараарай.

Хэрэгслийг ашиглахын тулд оролтын файл үүсгээд тухайлбал "sample1.in" эхний мөрөнд N байх ба дараагийн мөрөнд N тоонуудын сэлгэмэлээр тодорхойлоод өөр нэг мөрөнд нь N битүүдийг шувуунуудын анхны төлөвөөр тодорхойлно: Жишээлбэл:

```
6
1 2 0 4 3 5
1 1 0 0 1 0
```

Пайтон программд, `solution.py` (Хэвийн ажиллагаанд нь `python3 solution.py`):

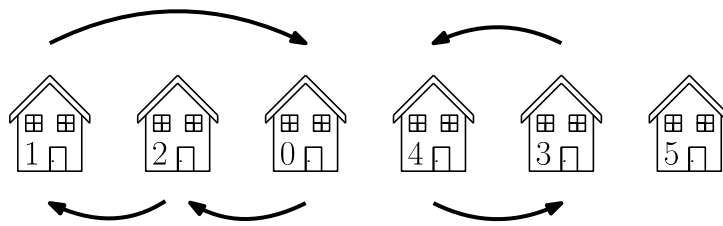
```
python3 testing_tool.py python3 solution.py < sample1.in
```

C++ программд, эхлээд хөрвүүлээд (Жишээ нь хамт `g++ -g -O2 -std=gnu++20 -static solution.cpp -o solution.out`) тэгээд ажиллуулна:

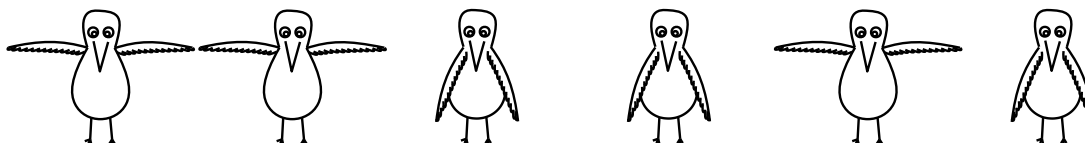
```
python3 testing_tool.py ./solution.out < sample1.in
```

Жишээ

Жишээн дээр байшингуудад дараах хүмүүсийн сэлгэмэл өгөгдөнө:



Жишээ программ эхний удаа $w = 0$ ажиллана, тэр нь гаралт нь $W = 2$ байна. Юу гэсэн үг вэ гэхээр Детжи гудамжаар хоёр удаа явна гэсэн үг. (программ хоёр удаа ажиллана) Эхний явалтын өмнө шувуунууд дараах байдалтай байна.



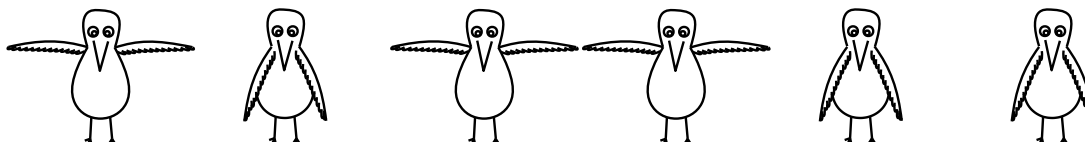
Тэгээд $w = 1$ үед программ ажиллана. Энэ нь Детжигийн эхний явалтыг илэрхийлнэ. Тэр нэг нэгээр нь шувуунуудаар явна. Зүүнээсээ эхлээд боломжтоор тэдгээрийн төлөвийг өөрчилнө. Жишээ программ биднийг $(i + 1)$ дэх шувууны төлөвийг харахаас өмнө i дахь шувууны төлөвийг гаргана.

Сургууль дээр Детже ирсний дараа шувуудын төлөв дараах байдалтай харагдана:



Сүүлийн программын ажиллагаанд ($w = 2$) Детжи сургуулиас гэртээ ирнэ. Энэ тохиолдолд тэр шувуунуудыг баруунаас зүүн тийш явахдаа харах ба ийм дарааллаар процессыг явуулна. Энэ нь $(i - 1)$ дэх шувууг харахаас өмнө i дэх төлөвийг тодорхойлох хэрэгтэй гэсэн үг.

Түүнийг явсны дараа шувуунууд доорх байлдаар харагдана:



Үнэндээ, энэ бол зөв тохиргоо. Жишээлбэл 3 шувууны төлөв бол дэлгээтэй. (өөрөөр хэлбэл зүүнээс дөрөв дэх) байна. (одоо $b_3 = 1$), 4 дугаартай оршин суугч энд нүүж ирэх ба ($a_3 = 4$) түүний анхны шувуу нь дэлгээтэй төлөвт байсан (анх $b_4 = 1$).

Шалгагч гаралт	таны гаралт
0 6	
1 2 0 4 3 5	
	2

Шалгагч гаралт	Таны гаралт
1 6	
1 2 0 4 3 5	
1	
	0
1	
	1
0	
	0
0	
	1
1	
	1
0	
	1

Шалгагч гаралт	Таны гаралт
2 6	
1 2 0 4 3 5	
1	
	0
1	
	0
1	
	1
0	
	1
1	
	0
0	
	1