

D. Bog' Dekoratsiyalari (Garden Decorations)

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

Har kuni maktabga va uyga qaytuvchi yo'lda yurar ekan, Detje ko'chada N ta uyning yonidan o'tadi, bunda uylar 0 dan $N - 1$ gacha raqamlangan. Hozirda i -raqamli uyda i -raqamli odam yashaydi. Shunchaki zerikishgani uchun, odamlar bir-biri bilan uylarni almashishga qaror qilishdi. Reja bo'yicha i -uyga $a[i]$ -raqamli odam ko'chib o'tadi (qaysiki avval $a[i]$ -uyda yashardi).

Barcha uylarning bog'ida qush haykalchalari bor. Haykalchalar ikki xil holatda bo'lishi mumkin: qanotlari *ochiq* (qush uchayotgandek), yoki qanotlari *yopiq* (qush yerda turgandek). Aholi qushlarning ko'rinishi borasida juda ham qattiq talabga ega. Hozirda i -uy to'g'risidagi qushning holati i -odamning o'zi xohlagan holatida turibdi. Odamlar o'zining yangi bog'idagi qushning ko'rinishi eski bog'idagi qushning ko'rinishi bilan bir xil bo'lmagunicha yangi uyga ko'chishga rozi bo'lishmaydi. Detje haykalchalar holatini to'g'irlash orqali aholini yangi uylariga ko'chishga yordam bermoqchi.

Buning uchun, u quyidagi ishni qiladi: qachonki u ko'chada sayr qilsa (maktabga borayotganda yoki uyga qaytayotganda), u yo'lidagi qushni o'rganadi va xohlasa uning holatini o'zgartirib qo'yadi (qanotini ochish yoki yopish orqali). Detjening maktabda va uyda qiladigan ishlari ko'p bo'lgani uchun, **u avvalgi sayrlaridagi qushlarning holatlarini eslab qolmaydi**. O'zining baxtiga, Detje a_0, a_1, \dots, a_{N-1} sonlarini yozib olgan, demak u qaysi odam qaysi uyga ko'chib o'tishini biladi.

Detjega qaysi qushlarni o'zgartirish strategiyasini tuzishga yordam bering, bunda qushlarning holatlari aholining istaklariga mos tushsin. U ko'chada ko'pi bilan 60 marta sayr qilishi mumkin, lekin yuqoriroq ball olish uchun, u ko'chada kamroq sayr amalga oshirish kerak.

Dasturlash

Bu multirun masala, ya'ni dasturingiz bir-necha marta ishga tushiriladi.

Barcha ishga tushirishlarda, siz birinchi bo'lib ikkita butun sonni qabul qilishingiz kerak, w va N , sayrning indeksi va uylar soni. Dasturingizning birinchi ishlatilishida $w = 0$ bo'ladi, ikkinchisida $w = 1$ va h.k. (ko'proq ma'lumotlar pastda berilgan).

Ikkinchi qatorda sizga N ta a_0, a_1, \dots, a_{N-1} butun sonlari kiritiladi, bu i -raqamli uyga a_i -raqamli uyda yashayotgan inson ko'chib o'tadi. a_i qiymatlar *permutatsiya* hosil qiladi: ya'ni, 0 dan $N - 1$ gacha barcha sonlar a_i da aynan bir marta uchraydi. E'tibor bering, qaysidir odam hech qayerga ko'chmaslikni tanlashi mumkin; ya'ni, $a_i = i$ bo'lishi mumkin.

Aholi o'zining uyini faqat bir marta o'zgartiradi. Ya'ni, qaysidir test uchun, N va a_i qiymatlar dasturingizning barcha ishga tushirishlarda bir xil bo'ladi.

Birinchi Ishlatish.

Dasturingizning birinchi ishga tushirilishida $w = 0$ bo'ladi. Bu ishlatishda, siz shunchaki W ($0 \leq W \leq 60$) butun sonini, ya'ni Detjening sayrlari sonini chiqarishingiz kerak. So'ngra dasturdan chiqishingiz kerak (exit). Shundan so'ng dasturingiz yana W marta ishga tushiriladi.

Keyingi Ishlatishlar.

Keyingi ishga tushirishda $w = 1$ bo'ladi; undan keyingisida $w = 2$; va h.k. so'nggi ishga tushirishda $w = W$.

Siz w , N , va a_0, a_1, \dots, a_{N-1} sonlarini qabul qilgangizdan so'ng, Detje ko'cha bo'ylab sayr qiladi.

- Agar w toq son bo'lsa, Detje uydan maktabga borayotgan bo'ladi, va uylarni $0, 1, \dots, N - 1$ tartibida ko'radi.

Dasturingiz keyingi qatorda b_0 ni qabul qilishi kerak, qaysiki 0 (yopiq), yoki 1 (ochiq) bo'ladi, bu 0-uyning to'g'risidagi haykalchanning holati. b_0 ni qabul qilgandan so'ng dasturingiz keyingi qatorda 0 yoki 1 chiqarishi kerak, b_0 ning siz xohlagan yangi qiymati.

So'ngra dasturingiz yangi qatorda b_1 qiymatini qabul qilishi kerak, bu 1-uyning to'g'risidagi haykalchanning holati; va yangi qatorda b_1 ning yangi qiymatini chiqarasiz. Bu narsa barcha N ta uy uchun takrorlanadi. So'nggi uydan o'tganingizdan so'ng, (ya'ni b_0 ning qiymatini kiritib/chiqarib bo'lganingizdan so'ng) **dasturdan chiqishingiz kerak.**

E'tibor bering, dasturingiz b_{i+1} ning qiymatini qabul qilishi uchun avval b_i ning yangi qiymatini chiqarishingiz kerak

- Agar w juft son bo'lsa, Detje maktabdan uyga qaytmoqda, va u uylarni $N - 1, N - 2, \dots, 0$ tartibida bosib o'tadi. Ya'ni siz avval b_{N-1} ning qiymatini kiritib/chiqarishingiz, keyin b_{N-2} , va h.k. yakunda b_0 qiymatini kiritib/chiqarishingiz kerak.

Agar $w = 1$ bo'lsa, kiruvchi ma'lumotlarda b_0, b_1, \dots, b_{N-1} qiymatlar haykalchalarning dastlabki holatini beradi. Agar $w > 1$ bo'lsa, kiruvchi ma'lumotlarda b_0, b_1, \dots, b_{N-1} qiymatlar dasturingiz avvalgi ishga tushirishda qaytargan qiymatlarga teng bo'ladi.

Yakunda, ya'ni so'nggi ishga tushirishdan so'ng, barcha i uchun b_i qiymatlar b_{a_i} ning dastlabki qiymatiga teng bo'lishi kerak, aks holda siz Wrong Answer verdiktini olasiz.

Detallar.

Agarda dasturingizning $W + 1$ ta ishga tushirishdagi ishlash vaqtlari *yig'indisi* vaqt limitidan oshib ketsa, yechimingiz Time Limit Exceeded verdiktini oladi.

Har bir qatorni chiqarganingizdan so'ng ma'lumotlar oqimini (standard output) ni bo'shatishni (flush) unutmang. Pythonda qatorlarni o'qish uchun `input()` ishlatsangiz bu ish avtomatik amalga oshadi. C++da, `cout << endl;` yangi qator chiqazish bilan birga oqimni bo'shatadi; agar `printf` ishlatsangiz, `fflush(stdout)` ishlating.

Cheklovlar va baholash

- $2 \leq N \leq 500$.
- Siz ko'pi bilan $W \leq 60$ ta sayr amalga oshira olasiz.

Yechimingiz bir nechta test guruhida tekshiriladi. Har bir test guruhi uchun bir necha ball belgilangan. Barcha test guruhlarida bir necha test bor. Test guruhi uchun ball olish uchun Siz ushbu guruhdagi barcha testlarga yechim berishingiz kerak.

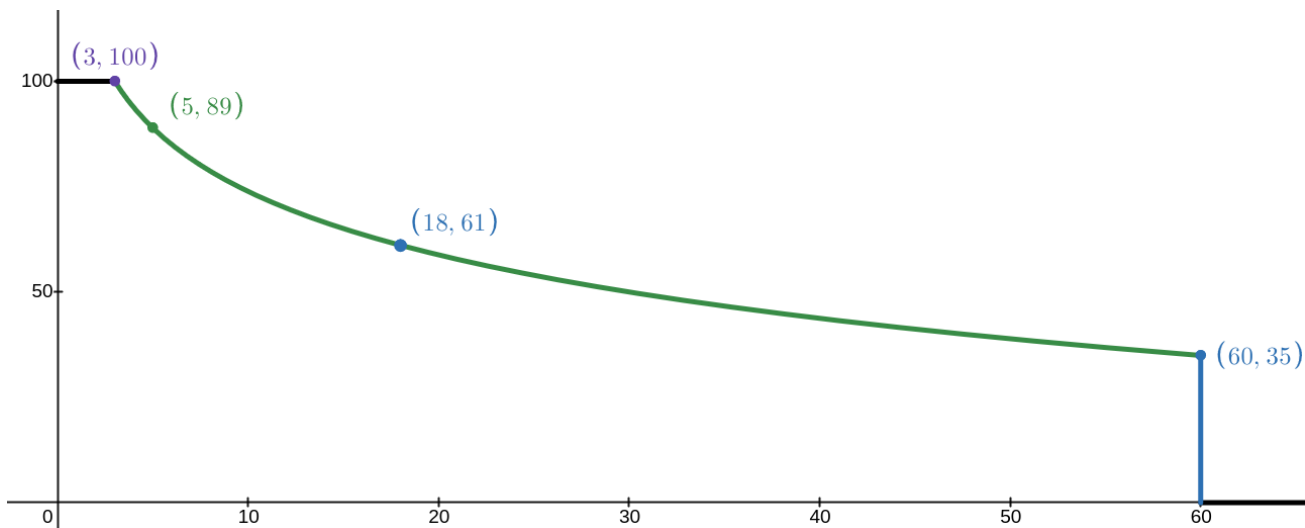
Guruh	Maks ball	Cheklovlar
1	10	$N = 2$
2	24	$N \leq 15$
3	9	$a_i = N - 1 - i$
4	13	$a_i = (i + 1) \bmod N$
5	13	$a_i = (i - 1) \bmod N$
6	31	Qo'shimcha cheklovlarsiz

Dasturingiz to'g'ri ishlagan barcha test guruhlari uchun, quyidagicha ball olasiz:

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

bu yerda S_g test guruhi uchun maksimal ballga teng, va W_g shu test guruhi testlarida ishlatilgan maksimal W ning qiymati. Barcha test guruhlarida olgan ballingiz eng yaqin butun songacha yaxlitlanadi.

Quyidagi diagramma oladigan ballingizni W ning funksiyasi orqali ifodalaydi. Umuman olganda, 100 ball olish uchun barcha testlarda $W \leq 3$ shart bajarilishi kerak.



Tekshirish qurilmasi

Yechimingizni tekshirishni osonlashtirish uchun, biz sizga oson qurilmani taqdim etamiz. Uni masalaning Kattis sahifasining pastki qismidagi "attachments" bo'limidan ko'chirib olishingiz mumkin. Bu qurilmani ishlatish ixtiyoriy. E'tibor bering, Kattisning haqiqiy tekshiruvchi greyderi ushbu qurilmadan farq qiladi.

Qurilmani ishlatish uchun, input fayl yarating, masalan "sample1.in", qaysiki N butun soni bilan boshlanishi, keyin esa N ta sonli permutatsiya, va yana boshqa qatorda boshlang'ich holatlarni beruvchi N ta bit (0 yoki 1) bilan davom etishi kerak. Misol uchun:

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

Python dasturlari uchun, aytaylik `solution.py` (odatda `python3 solution.py` buyruq orqali ishga tushirilad):

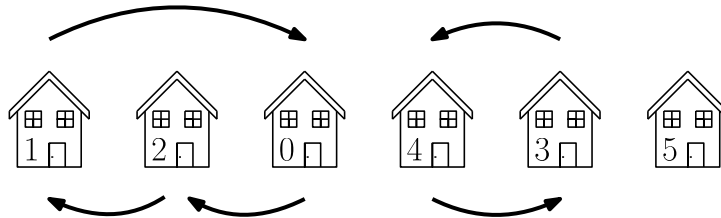
```
python3 testing_tool.py python3 solution.py < sample1.in
```

C++ dasturlari uchun, avval uni kompilyatsiya qiling: (misol uchun `g++ -g -O2 -std=gnu++20 -static solution.cpp -o solution.out` buyruq orqali) va so'ngra:

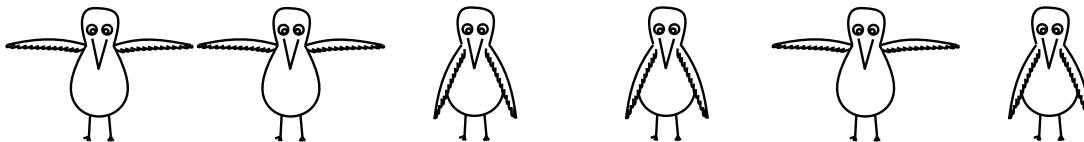
```
python3 testing_tool.py ./solution.out < sample1.in
```

Namunalar

Namunada, aholi permutatsiyasi quyidagi ko'rinishda:

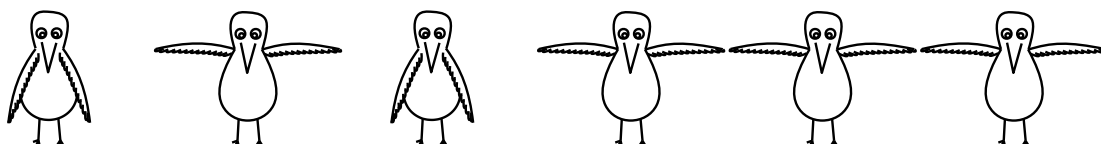


Birinchi marta namuna ishlatilganida ($w = 0$ bilan), dastur $W = 2$ chiqardi. Bu Detje sayrga ikki marta chiqishini bildiradi (va dastur yana ikki marta ishga tushiriladi). Birinchi sayrdan oldin bog'lardagi qushlarning holati quyidagicha:



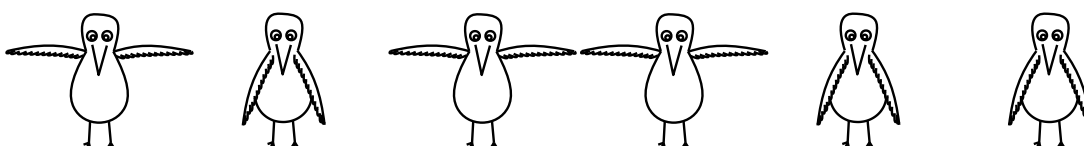
So'ngra dastur $w = 1$ bilan yana bir marta ishga tushiriladi: Detjening birinchi sayri. U qushlar bo'ylab birma-bir yurib chiqadi, va yo'l-yo'lakay qaysidir qushlar holatini o'zgartirishi mumkin. Dastur i -qushning yangi holatini ($i + 1$)-qushni ko'rishdan oldin chiqarishi kerak.

Detje maktabga kelganidan so'ng, qushlar quyidagi holatda bo'ladi:



Dasturingizning so'nggi ishga tushirilishida ($w = 2$ bilan), Detje maktabdan uyga qaytadi. Yodingizda tuting, u qushlarni o'ngdan-chapga tartibda ko'radi va o'sha teskari tartibda protsess qiladi! Bu degani u i -qushning yangi holatini ($i - 1$)-qushni ko'rishdan oldin aytishi kerak.

Detje sayrlarni tugatganidan keyin, qushlar quyidagi ko'rinishda bo'ladi:



Albatta, bu holatlar barchasi to'g'ri. Misol uchun, 3-qushning holati (ya'ni chapdan to'rtinchi) qanotlari ochiq (hozir $b_3 = 1$), va bu uyga 4-inson ko'chib keladi ($a_3 = 4$). Bu insonning eski uyidagi qushning holati qanotlari ochiq edi (boshlang'ich $b_4 = 1$).

grader output	your output
0 6	
1 2 0 4 3 5	
	2

grader output	your output
1 6	
1 2 0 4 3 5	
1	
	0
1	
	1
0	
	0
0	
	1
1	
	1
0	
	1

grader output	your output
2 6	
1 2 0 4 3 5	
1	
	0
1	
	0
1	
	1
0	
	1
1	
	0
0	
	1