

Daugybė legendų supa Pasvirusį Torunės bokštą. Bokšto sieną sudaro apskritimas su $N \geq 3$ durų, vienodais atstumais nutolusiomis viena nuo kitos (kitai tariant, durys yra taisyklingojo N -kampio viršūnės). Durys yra **atsitiktine tvarka** sužymėtos nuo 0 iki $N - 1$. Daugiau informacijos rasite skiltyje “Vertinimas”.

Viena iš mažiau žinomų legendų mena, kad kiekvienas naujas bokšto gyventojas turėjo įveikti išbandymą. Išbandymo tikslas – išvardinti visas duris vaikstant apskritimu (pagal arba prieš laikrodžio rodyklę), pradedant nuo bet kurių durų ir aplankant kiekvienas duris lygiai vieną kartą.

Tai turi būti atlikta nematant bokšto. Vietoj to naujasis gyventojas gali užduoti tokius klausimus: “Iš šių trijų skirtingų durų x, y, z , kurios yra tarpusavyje arčiausiai: $\{x, y\}$, $\{y, z\}$, ar $\{z, x\}$?”. Šios užklauskos atsakymas yra visos durų poros (tarp $\{x, y\}$, $\{y, z\}$ bei $\{z, x\}$) su mažiausiu Euklidiniu atstumu. Atstumas tarp durų yra lygus trumpiausios jas jungiančios atkarpos ilgiui. Jūsų užduotis – parašyti programą, kuri paklaususi nedaug užklauskų, galėtų surasti durų tvarką.

Užklauskos

Ši užduotis yra interaktyvi. Jūs turite parašyti programą, kuri rastų sprendimą siųsdama užklauskas ir gaudama atsakymus į jas per standartinį įvesties ir išvesties įrenginį.

Visų pirma jūsų programa iš standartinio įvesties įrenginio turi nuskaityti du sveikuosius skaičius t ir k ($1 \leq t \leq 100$, $1 \leq k \leq 12\,000$), atitinkamai žyminčius testų skaičių bei maksimalų vidutinį leidžiamą užklauskų skaičių (žr. skiltį “Vertinimas”).

Kiekvienam testui jūsų programa iš standartinio įvesties įrenginio turi nuskaityti vieną sveikąjį skaičių n ($3 \leq n \leq 500$), nurodantį durų skaičių bokšte.

Toliau jūsų programa turi užduoti klausimus tokiu formatu:

- Norėdama pateikti klausimą apie duris x, y bei z , jūsų programa į standartinį išvesties įrenginį turi išvesti:

? x y z

, kur x, y ir z yra skirtingi sveikieji skaičiai ($0 \leq x, y, z \leq n - 1$).

- Atsakymas bus pateiktas taip:

r
 a_1 b_1
...
 a_r b_r

, kur r yra sveikasis skaičius ($1 \leq r \leq 3$), aprašantis durų porų skaičių su mažiausiu atstumu. Kiekvieną tokią porą apibūdina du sveikieji skaičiai a_i ir b_i ($a_i, b_i \in \{x, y, z\}$ ir $a_i < b_i$).

Kai jau žinote durų tvarką, į standartinį išvesties įrenginį turite išvesti vieną atsakymo eilutę tokiu formatu:

! x_0 x_1 ... x_{n-1}

, kur x_0, x_1, \dots, x_{n-1} yra durų tvarka kaip aprašyta sąlygoje. Atkreipkite dėmesį, kad yra lygiai $2n$ galimų teisingų atsakymų, kadangi galite išvesti durų seką pradedami nuo bet kurių durų ir eidami apskritimu viena iš dviejų galimų kryptių. Visi šie atsakymai bus priimami.

Neužmirškite, kad po kiekvienos užklauskos reikia priverstinai išvalyti standartinio išvesties įrenginio buferį C++ naudojant `cout.flush()` (arba `fflush(stdout)`), jei naudojate `printf`) arba Python naudojant `sys.stdout.flush()`. Kitu atveju jūsų programa gali viršyti leistiną vykdymo laiką (gauti klaidą `Time Limit Exceeded`).

Išvedus atsakymą, jūsų programa privalo nedelsdama pradėti spresti kitą testą arba baigti darbą po visų testų įvykdymo.

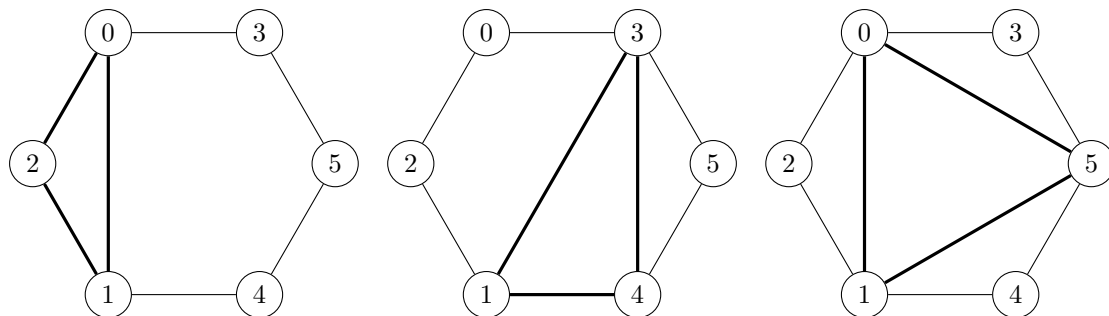
Programa negali skaityti jokių failų ar kitų operacinės sistemos resursų. Testavimo tikslais jūs galite papildomą informaciją išvesti į standartinį klaidų įrenginį (standard error stream), bet neužmirškite, kad rašymas į jį užtrunka papildomai.

Taip pat atkreipkite dėmesį, kad kiekvieno testo pradinė durų tvarka yra iš anksto fiksuota ir nesikeičia užklauskų metu.

Interaktyvaus sprendimo pavyzdys

Tarkime, kad turime tik vieną testą su $n = 6$, o durų tvarka yra 5, 3, 0, 2, 1, 4. Interaktyvus sprendimas galėtų atrodyti taip:

Interaktorius	Jūsų programa	Paiškinimas
1 100		$t = 1$ ir $k = 100$.
6		Pateikiamas pirmojo testo durų skaičius.
	? 0 1 2	Jūsų programa klausia, kurios durų poros atstumas mažiausias.
2 0 2 1 2		Durų poros $\{0, 2\}$ ir $\{1, 2\}$ yra arčiausiai.
	? 4 1 3	Jūsų programa klausia, kurios durų poros atstumas mažiausias.
1 1 4		Pora $\{1, 4\}$ yra arčiausiai.
	? 0 5 1	Jūsų programa klausia, kurios durų poros atstumas mažiausias.
3 0 5 0 1 1 5		Poros $\{0, 5\}$, $\{0, 1\}$, ir $\{1, 5\}$ yra arčiausiai.
	! 4 5 3 0 2 1	Jūsų programa teisingai išveda durų tvarką.



Pavyzdžio paaiškinimas: Aukščiau esančios iliustracijos vaizduoja duris ir jų numerius palei bokšto sienas. Pirmoje iliustracijoje iš kairės pavaizduotas trikampis, kurį sudaro durys 0, 1, 2, atitinkančios pirmą jūsų programos užklausą. Jame poros $\{0, 2\}$ ir $\{1, 2\}$ yra arčiausiai. Vidurinėje iliustracijoje pavaizduotas trikampis, kurį sudaro durys 1, 4, 3, atitinkančios antrą jūsų programos užklausą. Jame pora $\{1, 4\}$ yra arčiausiai. Paskutinėje iliustracijoje pavaizduotas trikampis, kurį sudaro durys 0, 1, 5, atitinkantis trečią jūsų programos užklausą. Jame visos durų poros yra nutolusios vienodu atstumu.

Atkreipkite dėmesį, kad sekos 0, 2, 1, 4, 5, 3 ar 5, 4, 1, 2, 0, 3 (bei keletas kitų) taip pat būtų priimtose šiame teste.

Vertinimas

Ši užduotis yra vertinama dalinėmis užduotimis. Kiekvieną dalinę užduotį sudaro tik vienas programos vykdymas ir šis vykdymas turi lygiai $t = 100$ testų. Kiekvienam vykdymui jūsų programos užduotų užklausų vidurkis suskaičiuojamas padalijant visų testų bendrą užklausų skaičių iš testų skaičiaus. Jei šis vidurkis yra didesnis negu k dalinei užduočiai, gausite 0 taškų už šią dalinę užduotį. Kitu atveju už dalines užduotis 1-4 gausite visus taškus.

Paskutinei dalinei užduočiai jūsų taškai bus suskaičiuoti taip: pažymėkime k^* jūsų programos užduotų užklausų vidurkį. Tada taškų kiekis suskaičiuojamas pagal šią formulę:

$$\left\lceil 56 \cdot \min \left(1, \frac{12000 - k^*}{7800} \right) \right\rceil$$

Tai reiškia, kad jūsų taškų skaičius tiesiškai didėja nuo 0 iki 56 mažėjant k^* nuo 12000 iki 4200.

Atkreipkite dėmesį, kad jei jūsų programa kuriam nors testui išveda neteisingą atsakymą, gausite 0 taškų už šią dalinę užduotį nepriklausomai nuo užduotų užklausų skaičiaus.

Papildomi ribojimai kiekvienai dalinei užduočiai aprašyti žemiau esančioje lentelėje.

Dalinė užduotis	Ribojimai	Taškai
1	$k = 8000, 3 \leq n \leq 9$	6
2	$k = 4500, 40 \leq n \leq 50$	7
3	$k = 3000, 90 \leq n \leq 100$	9
4	$k = 4500, n = 400$, egzistuoja teisinga išvestis x_0, \dots, x_{n-1} kur $x_i = i$ kiekvienam $200 \leq i \leq 399$	22
5	$k = 12\,000, n = 500$	iki 56

Be to, galite teigti, kad kiekvienas testas buvo sugeneruotas iš pradžių pasirenkant n **atsitiktinai pagal tolygų pasiskirstymą** iš visų galimų n reikšmių, tenkinančių dalinės užduoties ribojimus, ir tada pasirenkant durų tvarką **atsitiktinai pagal tolygų pasiskirstymą** iš visų galimų iš n durų tvarkų, tenkinančių dalinės užduoties ribojimus.