

Zadanie školského kola súťaže ZENIT v programovaní

Kategória A a B, 17.10.2023

V prípade nejasností konzultujte záložku **Pomoc** na stránke zenit.ksp.sk, alebo sa spýtajte organizátorov. Úlohy sú hodnotené úplne nezávisle a samostatne, takže ich môžete riešiť v ľubovoľnom poradí. Časový limit označuje, koľko času dostane váš program pri testovaní na našom serveri (nie na vašom lokálnom počítači). Počas súťaže môžete nájsť zadania aj na webstránke. Ak by sa papierové a tlačené zadania v nejakom detaile (napríklad časovom limite) nezhodovali, tak pravdu majú zadania na **webstránke**.

A: A ešte raz

12 bodov

Príde vlak načas? Alebo nie? Zuzka s tým nič nespraví, musí mať šťastie.

Aby sa v tom zlepšila, rozhodla sa trénovať v hode kockami. To je však celkom nuda, ak za tým nie je ďalšia motivácia.

V tom má však Zuzka šťastie - za hody kockami sú tento rok v Zenite body!

Úloha

Hod'te si dvoma kockami a sčítajte čísla, ktoré Vám padnú. Výsledok vypíšte na výstup a následne dostanete toľko bodov. Ak nie ste spokojný s výsledkom, môžete si hodiť znova.

Vstup a Výstup

Váš program nenačíta nič zo vstupu.

Vypíšte na jeden riadok súčet, ktorý Vám padol na kockách. Nevypisujte žiadny text navyše, iba toto číslo.

Príklad

Túto úlohu zoládnete aj bez príkladu :)

B: Budeš mojim rýchlikom?

15 bodov

Život slovenského rýchlika vôbec nie je jednoduchý. Nikto neocení, keď vzplanieš vášňou. Nikto ťa neuteší, keď sa utápaš žiaľom. Vnútri niekedy cítiš nevysvetliteľný mrazivý chlad (konkrétne v zime) alebo nevysvetliteľné neznesiteľné teplo (konkrétne v lete). Všetci na teba frflú a občas máš dokonca pocit, že ťa nikto nemá rád. Keby ťa naozaj chceli, tak im nevádi si na dobré veci počkať. Kludne aj 300 minút.

No jednu vec ti môžu všetci závidieť. Nech sa deje čokoľvek, niekde tam vonku je určite rýchlik, s ktorým tvoříš dvojicu. Keď ty vychádzaš zo svojej východziej stanice, z tvojej cieľovej stanice v približne rovnakom čase vychádza rýchlik, ktorý ide tú istú trasu, len opačne. Ak šťastie dá a nič nemešká (rozumej skoro nikdy), tak sa dokonca stretnete na polceste a vaši rušňovodiči si môžu zamávať.

Každý rýchlik má svoje číslo. Pre rýchliky, ktoré majú párne číslo platí, že ich dvojica má číslo o jedna väčšie. Pre rýchliky, ktoré majú nepárne číslo platí, že ich dvojica má číslo o jedna menšie. Napríklad taký rýchlik 614 tvorí pár s rýchlikom 615. Teraz, keď viete, ako to funguje, pre každý rýchlik, ktorý vám číslom zadáme viete povedať, akú má dvojicu a môžeme mu všetci spolu závidieť.

Úloha

Na vstupe dostanete číslo A reprezentujúce číslo rýchlika. Vašou úlohou je vypísať číslo rýchlika, ktoré s týmto rýchlikom tvorí pár.

Vstup a Výstup

Na jedinom riadku vstupu sa nachádza celé číslo A ($1 \leq A \leq 2 \cdot 10^5$).

Na jeden riadok vypíšte celé číslo zodpovedajúce rýchliku, ktorý s rýchlikom zo vstupu tvorí pár.

Príklady

vstup

614

výstup

615

vstup

615

výstup

614

C: Celkom dlhý vlak

20 bodov

Skúsený vedúci Matej kupuje miestenky pre deti na sústredenie. Ale zas a znovu stránka železníc robí blbosti a ukazuje mu vlak s absurdne veľa vozňami. Už sa mu to zopár krát stalo ale tento krát je to asi rekord. Vozňov je ale moc veľa na to, aby ich spočítal ručne. Preto sa rozhodol poprosiť vás o pomoc.

Úloha

Napíšte program, ktorý zistí koľko vozňov má vlak.

Vstup a Výstup

Na vstupe dostanete obrázok vlaku (štyri riadky).

Na výstup vypíšte jedno číslo - koľko vozňov má vlak. Vozňov bude maximálne 100.

Príklad

```

_ ] [ _ _ | o |
<-----| -1
/0-0-0

```

Toto je rušeň, nie vozeň.

0

vstup

```

-----
|0 0 0 0| |o| _ _ ] [ _
|-----| -|----->
o o 0-0-0\

```

výstup

1

vstup

```

_ ] [ _ _ | o | |0 0 0 0| |0 0 0 0|
<-----| -|-----| -|-----|
/0-0-0 o o o o

```

výstup

2

D: Dokonale diagonále omrvinky

24 bodov

Keď Kubo konečne našiel svoje sedadlo, zistil že ja ňom strašne veľa omrviniek. Všimol si, že omrvinky nie sú na sedadle len tak porozhadzované, ale sú usporiadané do štvorca. To Kuba tak zaujalo, že sa na omrvinky pozeral až do konca (teda kým vlak nedošiel na konečnú). Vyzerá to, že omrvinky sú dokonca dokonale diagonálne. Je to však naozaj tak?

Úloha

Napíšte program, ktorý zistí či sú omrvinky dokonale diagonálne.

Dve omrvinky sú na rovnakej diagonále, ak rozdiel riadku a stĺpca jednej omrvinky je rovnaký ako rozdiel riadku a stĺpca druhej. Napríklad omrvinka v druhom riadku a prvom stĺpci je na rovnakej diagonále ako omrvinka v štvrtom riadku a treťom stĺpci, lebo u oboch je rozdiel $2 - 1 = 4 - 3 = 1$. Však pozrite príklady :).

Omrvinky sú dokonale diagonálne, ak pre každú diagonálu platí, že na nej je práve jeden typ omrviniek.

Vstup a Výstup

Na prvom riadku vstupu bude jedno číslo $1 \leq n$ - veľkosť sedadla. Nasleduje n riadkov, na každom z nich je n medzerou oddelených čísel, popisujúce omrvinky na sedadle. Rôzne typy omrviniek sú reprezentované rôznymi celými číslami.

Na výstup vypíšte **dokonale diagonálne** ak sú omrvinky dokonale diagonálne, a **kopa smetia** ak nie je. Nezapomnite na znak konca riadku.

Obmedzenia

Budú dve sady, v prvej platí, že $n \leq 3$, v druhej $n \leq 1000$.

Príklad

vstup

```
5
1 2 3 4 8
5 1 2 3 4
4 5 1 2 3
7 4 5 1 2
2 7 4 5 1
```

výstup

```
dokonale diagonálne
```

vstup

```
3
1 2 3
4 5 6
7 8 9
```

výstup

```
kopa smetia
```

vstup

```
4
4 8 5 3
4 4 8 5
2 4 5 8
1 2 4 5
```

výstup

```
kopa smetia
```

vstup

```
3
4 5 6
5 6 8
6 8 4
```

výstup

```
kopa smetia
```

Mohlo by sa zdať že toto sú dokonale diagonálne omrvinky. Nenechajte sa pomýliť, naozaj to je iba kopa smetia.

E: Exaktne použiť kupóny

30 bodov

Maťko je vášnivý cestovateľ a milovník vlakov. Každý týždeň netrpezlivo otvára svoj obľúbený časopis Dopravca, pretože v niektorých vydaniach objaví špeciálne kupóny na vlakové lístky. Každý z týchto kupónov je platný presne na 5, 7, alebo 13 zástavok obojsmerne.

Tieto kupóny chce využiť na to aby si spravil výlet k svojim obľúbeným staniciam a lepšie si ich prehliadol. A keďže si okrem staníc chce pozrieť aj mestá v ktorých sú, tak ku každej z nich plánuje samostatný celodenný výlet. Napriek tomu že kupónov nazbieral už veľmi veľa, nechce nimi plytvať. Chce si byť istý, že každú svoju cestu môže zaplatiť presne jedným alebo viacerými z týchto kupónov, bez toho, aby mu zostali nevyužité zástavky. Ak by mal vystúpiť predčasne, radšej odloží výlet na neskôr.

Pomôžte Maťkovi zistiť, ktoré výlety dokáže zaplatiť presne pomocou kupónov na 5, 7 alebo 13 zástavok, a ktoré výlety sa nedajú pokryť týmito kupónmi.

Úloha

Napíšte program, ktorý pomôže Maťkovi zistiť, ktoré z jeho plánovaných výletov vie zaplatiť presne, a ktoré nie.

Vstup a Výstup

Prvé číslo na vstupe je $1 \leq n \leq 1\,000$ – počet výletov ktoré Maťko plánuje.

Nasleduje n riadkov. Na každom z nich je $1 \leq a_i \leq 10^9$ – počet zástavok, ktoré musí Maťko prejsť, aby sa dostal do cieľovej destinácie.

Pre každý z výletov vypíšte ANO, ak sa dá presný počet zastávok vyskladať z kupónov na 5, 7 alebo 13 zastávok. Ak to nie je možné, vypíšte NIE.

Polovičný počet bodov viete získať za program ktorý vyrieši vstupy kde všetky výlety sú vzdialené najviac 30 zastávok.

Príklad

vstup	výstup
<input type="text" value="4"/>	<input type="text" value="ANO"/>
<input type="text" value="5"/>	<input type="text" value="NIE"/>
<input type="text" value="6"/>	<input type="text" value="ANO"/>
<input type="text" value="10"/>	<input type="text" value="ANO"/>
<input type="text" value="12"/>	<input type="text" value=""/>

Matko dokáže prejsť presne 5 zastávok s jedným kupónom na 5 zastávok.

6 zastávok sa však nedá presne vyskladať.

Pre 10 zastávok použije dva kupóny na 5 zastávok.

12 zastávok vie vyskladať kombináciou kupónu na 5 a 7 zastávok.

F: Fakt sú rovnaké?

35 bodov

OPRAVA: v oboch príkladoch bola chyba, príklady sú opravené a riešenia budú pretestované

Kika ide zas na chatu do Valče. Tento krát ale nie na dovolenku. Treba vyčistiť stoku. Keďže nemá auto, tak sa rozhodla ísť vlakom. Ako je jej zvykom, na stanicu prišla minútu pred odchodom vlaku. Ako bežala na vlak, tak si všimla, že lístok drží dole hlavou, takže beží na vlak so zlým číslom. Keď ale lístok otočila správne, tak zistila, že je to rovnaké číslo. Prišlo jej to zaujímavé, tak sa rozhodla zistiť, koľko takých čísiel existuje.

Úloha

Vašou úlohou bude zistiť, koľko existuje k -ciferných čísiel, ktoré sú rovnaké, keď ich otočíte o 180 stupňov. Čífy vyzerajú nasledovne:

```
### # ### ## # # ### ## ### ## ##
# # # # # # # # # # # #
# # # ### ## ### ## ### # ### ##
# # # # # # # # # # # #
### # ### ## # ### ## # ### ##
```

Vstup a Výstup

Na vstupe je jedno číslo k ($1 \leq k \leq 20$).

Na výstup vypíšte jedno číslo – počet k -ciferných čísiel, ktoré sú rovnaké, keď ich otočíte o 180 stupňov.

Príklad

vstup	výstup
<input type="text" value="1"/>	<input type="text" value="5"/>
<i>0, 1, 2, 5 a 8</i>	
vstup	výstup
<input type="text" value="2"/>	<input type="text" value="6"/>
<i>11, 22, 55, 69, 88 a 96</i>	

G: Gombíky v rušni

45 bodov

Keďže vo vyspelých štátoch musia prevádzať vyspelé vlaky, a vyspelí ľudia vyspelých štátov čítajú knihy, je samozrejmosťou, že vyspelé vlaky vyspelých štátov disponujú vyspelými svetielkami na čítanie. Náš štát v snahe o vyspelosť sa teda rozhodol ku každému sedadlu nainštalovať svetielko.

Aby sa to zmestilo do rozpočtu, celé to spadá na brigádnikov Adama, Betku, Cyrila a Dávida. Traja z nich budú inštalovať svetielka do n vozňov, a štvrtý bude v rušni štučať gombíky a overovať, že svetielka idú keď majú. Keďže štvrtý je Dávid, v rušni bude on.

Adam, Betka a Cyril budú postupne inštalovať svetielka. Vozne sú očíslované od 1 po n . Najskôr sa Adam rozhodne, v ktorom vozni bude pracovať. Betka a Cyril sa chcú zdržiavať spolu, a tak chcú pracovať v susedných vozňoch, ideálne ďaleko od Adama. Aby to však nebolo podozrivé, nepôjdu príliš ďaleko - ak Adam pracuje vo vozni x , pôjde Betka do vozňa $2x$ a Cyril do vozňa $2x + 1$. Ak to nie je možné, budú protestovať.

Adamovi je ich stratégia jasná, a môže len gúľať očami. Rád by však brigádu dorobil čo najskôr, a tak by bol rád aby ste zistili, koľko im to bude trvať, ak si bude vyberať vozne optimálne.

Úloha

Vlak má n vozňov očíslovaných od 1 po n . V i -tom vozni treba nainštalovať v_i svetielok.

V jednej šichte pôjde Adam inštalovať svetielka do vozňa x , a Betka s Cyrilom pôjdu inštalovať svetielka do vozňov $2x$ a $2x + 1$. Adam nesmie vybrať také x , aby im to neumožnil. Ak niekto inštaluje svetielko vo vozni kde už žiadne svetielka na inštalovanie neostávajú, tak sa tam celú šichtu fláka. Počas jednej šichty nainštaluje každý z nich práve jedno svetielko vo vozni v ktorom je.

Za koľko najmenej šicht vedia naši brigádnici nainštalovať všetky svetielka? Dá sa to vôbec?

Vstup a Výstup

V prvom riadku je číslo n - počet vozňov vlaku. V druhom riadku je n medzerou oddelených celých čísel v_1, v_2, \dots, v_n - počet svetielok ktoré treba nainštalovať v jednotlivých vozňoch.

Vypíšte jedno číslo. Ak sa nedajú nainštalovať všetky svetielka podľa pravidiel popísaných v úlohe, vypíšte -1 . Inak vypíšte najmenší počet šicht, za ktorý sa to dá.

V prvej sade $1 \leq n, v_i \leq 1000$.

V druhej sade $1 \leq n \leq 1000$ a $1 \leq v_i \leq 10^9$.

V tretej sade $1 \leq n \leq 10^5$ a $1 \leq v_i \leq 10^9$.

Príklady

vstup

3
1 2 3

výstup

3

Adam pôjde tri krát do vozňa 1, a tak Betka s Cyrilom pôjdu tri krát do vozňa 2 a 3, a všetky svetielka sa im podarí nainštalovať.

vstup

2
1 2

výstup

-1

Ak si Adam vyberie vozeň 1, Cyril by chcel pracovať vo vozni 3, ale taký nieje. Ak si Adam vyberie vozeň 2, bude už protestovať aj Betka, lebo vo vlaku nie je ani vozeň 4 a ani vozeň 5.

vstup

7
2 5 3 1 4 8 9

výstup

15

H: Hor sa do sveta

50 bodov

Ocko Rušeň a mamka Koľajnica majú spolu veľa detí-vláčat.

Všetky vláčatá už odrástli, a vybrali sa do sveta. S rodičmi sú stále v kontakte, a vždy sa im ozvú, keď na svojich vandrovkách dorazia na ďalšiu stanicu.

Rodičia si to všetko pozorne zapisujú do zápisníka, a vedia o každom vláčati, akými stanicami si prešlo.

K tomuto zápisníku sa nedávno dostal Ujo Vozeň, ktorý ich chce na štyridsiate siedme výročie ich svadby prekvapiť darčekom: mapou sveta, so všetkými stanicami ktoré navštívilo nejaké ich vláčča.

Samotná mapa so stanicami sa kreslí ľahko - trasy vláčat sa nikde nekrižujú ani nespájajú. Problém sú názvy staníc. Každú z nich bude musieť Ujo Vozeň vlastnoručne vystrihnúť a nalepiť na mapu.

Keby aspoň vedel, koľko roboty ho čaká...

Úloha

Do sveta sa vybralo n vláčat.

Trasa každého vláčaťa je popísaná jedným reťazcom, určujúc postupnosť navštívených staníc. Názvy staníc sú neprázdne a pozostávajú z malých písmen anglickej abecedy. Názvy jednotlivých staníc v postupnosti sú oddelené znakom /.

Všetky vláčatá začali na domácej stanici `home`, ktorá je implicitná, a teda si ju Rušeň s Koľajnicou do zápisníka nepísali, a ich trasy začínajú / a názvom prvej stanice, ktorú potom navštívili.

Stanice s rovnakým názvom sú odlišné stanice práve vtedy, keď sú ich trasy z domu odlišné.

Zistite, koľko rôznych staníc bolo navštívených.

Vstup a Výstup

V prvom riadku je číslo $1 \leq n \leq 10^5$ - počet vláčat.

V každom z nasledujúcich n riadkov je jeden reťazec, popisujúci trasu jedného vláčaťa.

Súčet dĺžok všetkých trás na vstupe bude v prvej sade najviac 1000, a v druhej najviac 10^6 .

Vypíšte jedno číslo: počet rôznych staníc, ktorými vláčatá prešli.

Príklady

vstup

```
3
/bratislava/prievidza
/bratislava/pezinok
/dolnykubin/hornykubin/prievidza
```

výstup

```
6
```

Ujo Vozeň musí vystrihnúť nasledovné názvy: dolnykubin, hornykubin, bratislava, pezinok, prievidza, prievidza. bratislava stačí len raz, keďže prvé aj druhé vláča prešli tou istou stanicou. Prievidze sú však rôzne stanice, keďže k jednej je trasa cez bratislavu, a v druhej cez kubíny.

vstup

```
4
/zenit/zenit/zenit
/zenit/zenit/ksp
/ksp/zenit/ioi
/ksp/zenit/ioi
```

výstup

```
7
```

Prvé dve zenity prvých dvoch vláčat sú rovnaké stanice, ako aj všetky stanice posledných dvoch. Zvyšok sú všetko unikátne stanice.

I: Inšpektor meškaní: Kde sú vlaky?

65 bodov

Nitramovi Šullebovi, vedúcemu PR oddelenia nemenovanej železničnej spoločnosti ide o miesto. Práve totiž dovolal s riaditeľom. A ten mu povedal, že ak tento týždeň už konečne nevydá článok, letí!

Nitram však vôbec nevie čo sa deje, týždeň ani len nebol v kancelárii. Už začínal byť zúfalý a prezerat' pracovné pozície v Lidli, vtom však na to prišiel. Ved' použije evergreen vzorec - znižujúce sa meškania vlakov!

No to je síce pekné, ale klesajú vôbec tie meškania? Ktovie, ale to sa jednoducho vyrieši - napíše, že zberná perióda skončila v deň, keď práve klesali (alebo aspoň nerástli). Nemôže si však vybrať akýkoľvek taký deň - keby meškania klesali aj po tomto dni, riaditeľ by videl že vedúci PR si nevie ani len vyfabrikovať najlepšie dáta a to by bol už isto Nitramov koniec.

Nitram vie zistiť priemerné meškanie v akýkoľvek deň jedinou otázkou do databázy. Avšak z kryptických dôvodov má daná tabuľka 1500 stĺpcov¹ a tak jediná query trvá hodiny. Kedy bude Nitram nastupovať za pokladňu?

Úloha

V databáze sa nachádzajú informácie o n dňoch a priemernom meškaní vlakov počas každého z nich. Databázy sa môžete pýtať na meškanie v ľubovoľný deň, avšak môžete položiť najviac k takýchto otázok. Nájdite deň, kedy meškanie bolo menšie alebo rovné meškaniam v dňoch pred ním a po ňom. Ak má daný deň iba jedného suseda, tiež môže byť odpoveďou, stačí aby spĺňal nerovnosť s existujúcim susedom.

Vstup a Výstup

Prvý riadok vstupu obsahuje číslo $n(1 \leq n \leq 10^4)$, ktoré označuje počet dní v databáze. Na druhom riadku sa nachádza číslo k - maximálny počet otázok, ktoré sa môžete databázy spýtať.

¹<https://jimnymiller.github.io/ugliest-beautiful-codebase>

Databáze môžete zadávať otázky vypísaním riadku vo formáte $? x$, kde x je číslo dňa, ktorý vás zaujíma. Ako odpoveď testovač vypíše jeden riadok s jedným prirodzeným číslom do 10^9 – meškaním v daný deň. Pozície v poli sú indexované číslami od 1 do n .

Na vypísanie odpovede vypíšete riadok formátu $! x$, kde x je pozícia lokálneho nestriktného minima. Ak minimum existuje viacej, vypíšete ľubovoľné z nich. Ak žiadne neexistuje, vypíšete $! -1$.

Keďže táto úloha je interaktívna, po vypísaní každej otázky je nutné flushovať. V pythone sa to robí parametrom `print(..., flush = True)`, v C++ pomocou `cin.flush()`. Ak tak neurobíte, váš program môže skončiť verdiktom TLE.

Po vypísaní odpovede odpovede musí váš program skončiť. Taktiež musí skončiť, ak na niektorú otázku dostane odpoveď -1 , čo indikuje neplatnú otázku (napríklad nedodržený formát či prekročený počet otázok). Ak tak neurobí, môžete dostať nesprávny verdikt.

Obmedzenia

Úloha má 5 sád vstupov. V jednotlivých sádach platia nasledujúce obmedzenia pre k :

Sada	1	2	3	4	5
$\$ k = \$$	10^4	5000	80	35	28

Príklady komunikácie

```
10000
10000
? 123
234
? 122
234
? 121
572
! 122
```

Prvok na pozícii 122 je menší alebo rovný obom svojím susedom, je riešením úlohy.

```
10000
80
? 1
4
? 2
9
! 1
```

Prvý prvok poľa je menší ako druhý, ktorý je jeho jediným susedom, teda je lokálnym minimom.

J: Jankine lístky

75 bodov

Janka už nie je študent. Ako neštudent už nemá právo na lístky na vlak zadarmo. A to je problém, pretože podľa budgetu (ktorý si tento týždeň usilovne pripravila), jej už neostane na knihy (ktoré sú na Slovensku luxusný tovar) od jej obľúbeného autora (ešte o tom nevie, ale je to Brandon Sanderson).

Našťastie má Janka aj riešenie. Všimla si, že keď si kúpite lístok z Bratislavy do Trenčína, lístok z Trenčína do Žiliny a lístok zo Žiliny do Kraľovian, tak v súčte tieto lístky stoja menej ako keď si kúpite lístok priamo z Bratislavy do Kraľovian. Ak ju niekedy stretnete, môžete sa jej spýtať, ako to zistila.

Teraz by Janku zaujímalo, či dokáže podobným spôsobom ušetriť aj na iných úsekoch jej obľúbenej trasy. Potrebuje totiž kúpiť maslo.

Úloha

Pri tejto úlohe skúmame ceny dopravy na jednej vlakovej linke v jednom smere.

Na vstupe dostaneme číslo N označujúce počet staníc na vlakovej linke. Stanice sú číslované od 1 (východisková stanica) po N (konečná stanica). Ďalej dostanete pre každú dvojicu staníc cenu lístka medzi nimi (v jednom smere).

Vašou úlohou bude pre M rôznych dvojíc staníc zistiť, koľko stojí najlacnejšia cesta medzi nimi.

Nie vždy platí, že priamy lístok medzi stanicami a a b je najlacnejšia voľba. Môže sa totiž stať, že keď nejaký úsek rozbijeme na podúseky, súčet cien lístkov na týchto podúsekoch je menší ako lístok pre celý úsek.

Niekedy najoptimálnejšia cena môže dokonca zahŕňať lístky, ktoré sa prekrývajú alebo zasahujú mimo skúmaný úsek. Napríklad si kúpite lístok zo stanice 2 do 5 a lístok zo stanice 4 do 9 a je to najlacnejšie možné kombo na cestovanie zo stanice 3 do 8.

Vstup a Výstup

Prvý riadok vstupu obsahuje kladné celé číslo N - počet miest.

Nasledujúcich $N - 1$ riadkov popisuje ceny lístkov: k -ty z týchto riadkov obsahuje $N - k$ medzerou oddelených celých čísel $c_{k,l}$ ($k < l \leq N$). Tieto čísla reprezentujú ceny priamych lístkov zo stanice k do každej z nasledovných staníc. Platí, že $0 < c_{k,l} < 2000000000$.

Ďalší riadok obsahuje kladné celé číslo M - počet dopytov.

Nasledujúcich M riadkov popisuje jednotlivé dopyty: i -ty z týchto riadkov obsahuje dve medzerou oddelené celé čísla a_i a b_i ($1 \leq a_i < b_i \leq N$), reprezentujúce začiatočnú a cieľovú stanicu.

Na výstup vypíšte M riadkov, každý obsahujúci jedno kladné celé číslo r_i - najmenšiu možnú cenu za lístky, ktoré nám pokryjú celú cestu zo stanice a_i do stanice b_i .

Body za túto úlohu sú rozdelené do 5 sád po 15 bodov podľa nasledovných kritérií:

1. $N = 15$, $M = 1$, pýtame sa na celý úsek a je zaručené, že v optimálnej trase sa lístky neprekrývajú
2. $N = 15$, $M = 1$, pýtame sa na celý úsek
3. $N = 40$, $M = 780$
4. $N = 1500$, $M = 5$
5. $N = 350$, $M = 61075$

Príklad

vstup

```
4
22 31 60
14 28
30
3
1 4
1 3
3 4
```

výstup

```
50
31
28
```

Pri prvom dopyte sa nám oplatí kúpiť lístok zo stanice 1 do stanice 2 za 22 a potom lístok z 2 do 4 za 28.

Pri druhom dopyte je najlacnejší priamy lístok z 1 do 3 za 31.

Pri treťom dopyte by nás priamy lístok z 3 do 4 stál 30, ale ak si kúpime dlhší lístok z 2 do 4, bude nás stáť menej.

vstup

```
5
11 18 27 40
7 16 29
9 22
13
4
1 5
2 4
1 3
3 5
```

výstup

```
40
16
18
22
```

V tomto prípade sú ceny lístkov konzistentné, takže je nám úplne jedno, ako si lístky kupujeme.

K: Kreslenie

85 bodov

Erikire má divné meno a divnú obsesiu. V každom aspekte svojho života sa snaží udržiavať symetriu, či už tým, že nosí ofinu a cop, nikdy nespí na boku, alebo že pri jedení používa dve vidličky. Erikiremu sa veľmi páčia nové osobáky, keďže majú nielen dvere na oboch stranách vozňov (na rozdiel od autobusov), ale aj ťažnú súpravu na oboch koncoch vlaku (na rozdiel od rýchlíkov). Erikire všade cestuje osobnými vlakmi a túži si svoje preferované trasy (ktoré majú na oboch stranách trate presne rovnako veľa stromov) schematicky zakresliť do plánárnej mapy. Táto mapa samozrejme musí spĺňať určitú vlastnosť.

Úloha

Trasy, po ktorých Erikire cestuje, sú tvorené mestami a traťami, ktoré ich spájajú. Spolu tvoria súvislý acyklický graf. Erikire z nich veľmi túži nakresliť schematickú traťovú mapu. Tá by mala mať na vrchu jeho bydlisko a všetky ostatné mestá niekde pod ním. Trate, ktoré spájajú mestá, sa nesmú krížiť a mapa musí byť samozrejme symetrická. Zistite, prosím, či sa to dá alebo nedá spraviť, nech už o tom Erikire konečne prestane rozprávať.

Vstup a Výstup

Prvý riadok obsahuje číslo N – počet miest ($1 \leq N \leq 10^5$). Nasleduje N riadkov. Na i -tom riadku je $k_i + 1$ medzerou oddelených čísel. Prvé číslo bude k_i , ktoré udáva počet susedov vrcholu i . Zvyšných k_i čísel sú čísla vrcholov jeho susedov. Erikireov dom má vždy číslo 0 a indexujeme od 0.

Vypíšte jeden riadok s textom **yeey**, ak sa dá nakresliť požadovaná mapa, inak **noo**.

Príklad

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

Ak umiestnime mesto číslo 3 a 4 do stredu, mesto 1 doľava a mesto 2 doprava, dostaneme symetrickú mapu.

yeey

vstup

```
7
3 1 2 3
1 0
1 0
3 0 4 5
1 3
2 3 6
1 5
```

výstup

noo

vstup

```
11
2 1 2 9
3 0 3 4
3 0 5 6
2 1 7
1 1
1 2
1 2 8
1 3
1 6
2 0 10
1 9
```

výstup

yeey

L: Lereimáš a Miško

90 bodov

Táto úloha je rovnaká ako úloha zenit23ck2j

Raz sa malý Miško zúčastnil slovného hádania so svojimi kamarátmi. Všetci sedeli okolo stola a snažili sa uhádnuť tajné heslo. Keď prišla Miškova poradová chvíľa, bol nervózny (pochopiteľne). No namiesto toho, aby povedal nejaké slovo, začal koktať a vypúšťať nahodilé hlásky.

“Kááá... éééé... ešte... mmmm... ttt...,” koktal Miško beznádejne.

“Kemť?” opýtal sa neveriacky Lereimáš. Všetci sa zasmiali a za bruchá pochytili, pretože *kemť* nebolo ani zďaleka správne heslo. No Miško sa nevzdal a pokračoval v hádaní nahodilých písmen.

“Možno by bolo lepšie, keby si len hádal slová, Miško,” povedal mu jeden z kamarátov s úsmevom. “Tvoja mama,” odpovedal Miško s úsmevom a pokračoval v hádaní.

Úloha

Miško postupne hovorí nahodilé písmená z abecedy až kým súvisle nevysloví celé heslo. Miško vyslovuje písmená nezávisle a rovnomerne náhodne. Aký je očakávaný počet písmen, ktoré Miško vysloví, kým sa mu konečne podarí vysloviť celé heslo?

Vstup a Výstup

Prvý riadok vstupu obsahuje číslo A , ktoré označuje veľkosť abecedy, z ktorej Miško vyslovuje písmená.

Na druhom riadku je heslo H , pozostávajúce iba z prvých A malých písmen anglickej abecedy.

Vypíšte jedno desatinné číslo, ktoré označuje očakávaný počet vyslovených písmen, ktoré Miško vysloví, kým nepovie súvisle heslo H .

Keďže počet písmen môže byť veľmi veľký, vypíšte ho modulo $10^9 + 7$ (teda celočíselnú časť zmodulujte a desatinnú nechajte tak). Odpovede sú považované za správne, ak je ich absolútna *alebo* relatívna chyba nanaajvyš 10^{-6} .

Sada	1	2	3	4	5	6	7,8	9,10	11,12	13,14,15
$1 \leq \ H\ \leq$	5	5	5	10	10	10	20	50	100	300
$1 \leq A \leq$	2	3	5	2	3	10	10	10	10	26

Príklad

vstup	výstup
<div style="border: 1px solid black; padding: 5px;">2 aaa</div>	<div style="border: 1px solid black; padding: 5px;">14.0</div>
vstup	výstup
<div style="border: 1px solid black; padding: 5px;">2 abb</div>	<div style="border: 1px solid black; padding: 5px;">8.0</div>
vstup	výstup
<div style="border: 1px solid black; padding: 5px;">3 abbc</div>	<div style="border: 1px solid black; padding: 5px;">81.0</div>