Hrvatska informatička olimpijada

Zagreb, 25. ožujka 2018.

Zadaci

Ime zadatka	Vremensko ograničenje	Memorijsko ograničenje	Broj bodova
Paprike	1 sekunda	1024 MiB	100
Pick	1 sekunda	1024 MiB	100
Svjetlost	3 sekunde	1024 MiB	100
Zagonetka	3 sekunde	1024 MiB	100
Ukupno			400

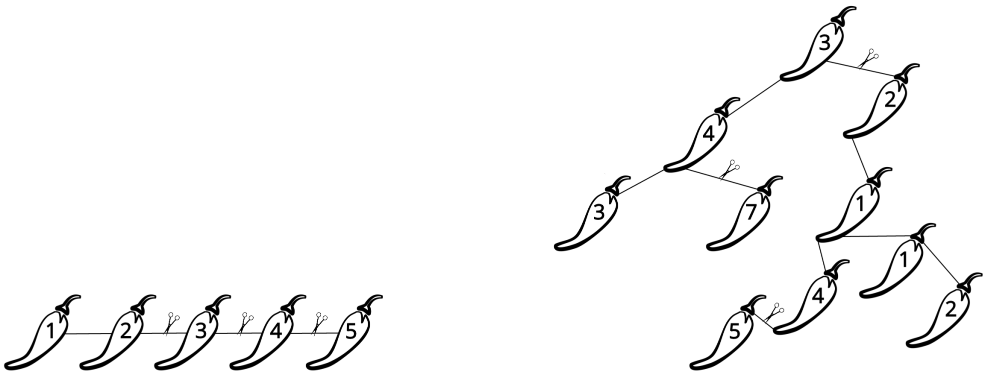
ministarstvo znanosti i obrazovanja

HRVATSKI SAVEZ INFORMATIČARA

HRVATSKA ZAJEDNICA TEHNIČKE KULTURE

Zadatak: Paprike

Krešo je na lokalnom obiteljsko-poljoprivrednom gospodarstvu kupio hrpu ljutih papričica uredno povezanih komadima špage u takozvani *vijenac*. U ovom zadatku, vijenac se sastoji od n paprika i $(n-1)$-og komada špage. Svaki komad špage povezuje dvije različite paprike te su svake dvije paprike u vijencu (direktno ili indirektno) povezane špagom. Dakle, paprike i komadi špage čine takozvano *stablo*. Jednim potezom škara Krešo može prerezati jednu špagu te vijenac paprika rastaviti na dva manja vijenca, koja opet rezovima može rastaviti na manje vijence itd. Primijetite da jedna ni sa čim povezana paprika također čini vijenac.



Slika 1: Početni vijenci iz prva dva primjera test podataka zajedno sa optimalnim rezovima.

Ljutina pojedine paprike mjeri se pomoću takozvane Scovillove ljestvice te je izražena nenegativnim cijelim brojem. Ljutina vijenca je suma ljutina pojedinih paprika koje sadrži. Krešo želi zapapriti ručak srednjoškolcima nakon informatičkog natjecanja i zna da prosječni srednjoškolac može pojesti vijenac ljutine najviše k prije nego što mora zatražiti pomoć liječnika i dječjeg pravobranitelja.

Odredite najmanji broj rezova potreban da Krešo rastavi početni vijenac na vijence ljutine najviše k.

Ulazni podaci

U prvom redu nalaze se cijeli brojevi n i k — broj paprika i najveća dopuštena ljutina jednog vijenca. Paprike su označene brojevima od 1 do n. Sljedeći red sadrži n cijelih brojeva $h_1, h_2, \ldots, h_n$ — broj h_j je ljutina paprike j. Svaki od sljedećih $n-1$ redova sadrži dva različita prirodna broja x i y $(1 \leq x, y \leq n)$ — oznake paprika direktno povezanih špagom u početnom vijencu. Paprike i špage čine stablo kao što je opisano u tekstu zadatka.

Izlazni podaci

Ispišite traženi najmanji broj rezova.

Bodovanje

U svim podzadacima vrijedi $n \geq 2$ i $0 \leq h_1, h_2, \ldots, h_n \leq k \leq 3\,000\,000$.

Podzadatak	Broj bodova	Ograničenja
1	11	$n \leq 15$
2	13	$n \leq 100\,000$, svaka paprika $x = 1, \ldots, n-1$ je povezana s paprikom $x+1$.
3	27	$n \leq 1\,000$
4	49	$n \leq 100\,000$

Primjeri test podataka

ulaz

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

izlaz

```
3
```

ulaz

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

izlaz

```
3
```

ulaz

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

izlaz

```
2
```

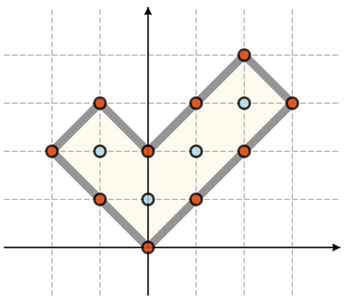
Zadatak: Pick

Mirko je nedavno čitao o *Pickovom teoremu* koji kaže sljedeće: ako u koordinatnom sustavu nacrtamo poligon čiji su vrhovi točke s cjelobrojnim koordinatama te ako A označava njegovu površinu, i broj točaka s cjelobrojnim koordinatama unutar poligona, a b broj točaka s cjelobrojnim koordinatama na njegovim stranicama (uključujući i vrhove poligona) onda uvijek vrijedi:

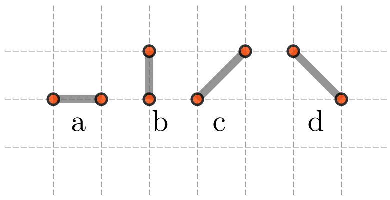
$$A = i + \frac{b}{2} - 1.$$

Kako bi testirao teorem, Mirko je na svojoj pametnoj ploči napravio poligon od magnetnih štapića koji su pod utjecajem gravitacije po noći skliznuli na dno ploče. Sada Mirko želi konstruirati poligon najmanje moguće površine tako da upotrijebi sve pronađene štapiće. Mirko štapiće može translatirati bilo gdje na svoju ploču, ali ih ne smije rotirati. Mirko na raspolaganju ima:

- a horizontalnih štapića duljine 1,
- b vertikalnih štapića duljine 1,
- c dijagonalnih štapića duljine $\sqrt{2}$ koji zatvaraju kut od 45° s pozitivnim dijelom x-osi,
- d dijagonalnih štapića duljine $\sqrt{2}$ koji zatvaraju kut od 135° s pozitivnim dijelom x-osi.



Slika 2: Za poligon vrijedi $A = 8, i = 4, b = 10$.



Slika 3: Štapići koje imamo na raspolaganju.

Odredite poligon najmanje moguće površine koji se može dobiti tako da se iskoriste svi štapići. Možete pretpostaviti da su ulazni podaci takvi da je moguće konstruirati barem jedan poligon sa zadanim štapićima.

Također, moguće je osvojiti djelomične bodove ako, koristeći sve zadane štapiće, konstruirate neki ispravan poligon (koji nije nužno najmanje moguće površine). Za više detalja pogledajte poglavlje “Bodovanje”.

Ulazni podaci

U prvom redu nalaze se četiri cijela broja a, b, c, d iz teksta zadatka.

Izlazni podaci

Ispišite n redova gdje je $n = a + b + c + d$. U j-ti red ispišite cijele brojeve x_j i y_j — koordinate j-tog po redu vrha poligona. Prvi vrh poligona mora biti $(0, 0)$, a ostali vrhovi mogu biti ispisani u proizvoljnom smjeru (pozitivnom ili negativnom). Dozvoljeno je da uzastopne strane poligona budu paralelne, ali poligon ne smije doticati niti sjeći samoga sebe.

Bodovanje

U svim podzadacima vrijedi $0 \leq a, b, c, d \leq 100$ i $a + b + c + d \geq 3$.

Podzadatak	Broj bodova	Ograničenja
1	5	$c = d = 0$
2	5	$a = b = 0$
3	10	$a + b + c + d \leq 6$
4	10	$a + b + c + d \leq 20$
5	10	$a + b + c + d \leq 40$
6	10	$a + b + c + d \leq 80$
7	10	$a + b + c + d \leq 150$
8	10	$a + b + c + d \leq 200$
9	10	$a + b + c + d \leq 300$
10	20	nema dodatnih ograničenja

Ako vaš program za neki test podatak ne ispiše ispravan poligon koji se sastoji od zadanih štapića, onda on dobiva 0 bodova za odgovarajući podzadatak. Ako program ispiše ispravan poligon koji nije najmanje moguće površine, onda može dobiti djelomične bodove prema sljedećim pravilima.

Za test podatak j, označimo s r_j omjer između površine ispisanoga poligona i najmanje moguće površine poligona. Za podzadatak k, označimo sa z_k najveći od brojeva r_j gdje je j test podatak iz podzadatka k. *Postotak* bodova P_k koji rješenje osvaja za podzadatak k ovisi o broju z_k na sljedeći način: $P_k = 10$ ako je $z_k \geq 3$, a inače se računa po formuli:

$$P_k = \frac{25}{8}(3 - z_k)^4 + 10.$$

Dakle, rješenje koje nije optimalno može osvojiti između 10% i 60% bodova za određeni podzadatak, ovisno o omjeru površine pronađenog i optimalnog poligona.

Primjeri test podataka

ulaz

```
1 1 1 0
```

izlaz

```
0 0
1 1
0 1
```

ulaz

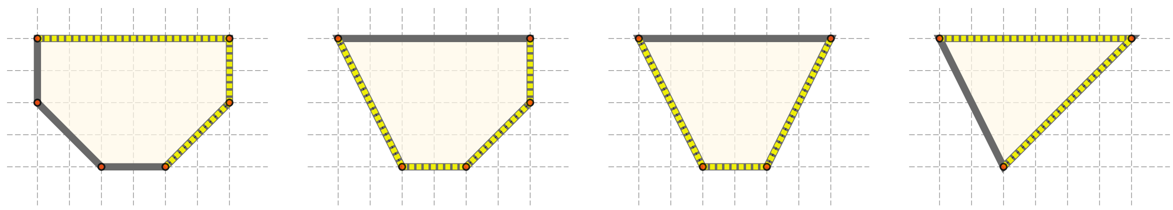
```
0 0 6 4
```

izlaz

```
0 0
1 1
2 2
3 3
2 4
1 3
0 2
-1 3
-2 2
-1 1
```

Zadatak: Svjetlost

Ako se u ravnini nalazi konveksni poligon P, a u neku točku T izvan poligona postavimo izvor svjetlosti, onda taj izvor osvjetljava neke bridove od P — ako su A i B dva uzastopna vrha poligona, onda je brid $\overline{AB}$ *osvijetljen* ako trokut $\triangle TAB$ nije površine nula te ne siječe unutrašnjost poligona. *Osvijetljenost* poligona je suma duljina svih osvijetljenih bridova, a *maksimalna osvijetljenost* poligona je najveća moguća osvijetljenost koju možemo postići ako na optimalan način odaberemo točku T. Pritom, točka T može biti proizvoljno udaljena od poligona te ne mora imati cjelobrojne koordinate.



Slika 4: Poligoni P, P_1, P_2 i P_3 iz drugog primjera test podataka, naznačeno je optimalno osvjetljenje.

Zadan je konveksni poligon P čiji su vrhovi redom točke $A_1, A_2, \ldots, A_n$. Poligon mijenjamo kroz q koraka — u j-tom koraku obrišemo jedan postojeći vrh poligona te tako dobijemo novi poligon P_j. Točnije, vrhovi poligona P_j su dosad neobrisani vrhovi početnog poligona P te dolaze istim redoslijedom kao u poligonu P. Lagano se vidi da je svaki poligon P_j opet konveksan.

Odredite maksimalnu osvijetljenost početnog poligona P te svakog od dobivenih poligona P_1, P_2, $\ldots$, P_q.

Ulazni podaci

U prvom redu nalazi se prirodan broj n — broj vrhova početnog poligona P. U j-tom od sljedećih n redova se nalaze dva cijela broja x_j i y_j $(-10^9 \le x_j, y_j \le 10^9)$ — koordinate vrha A_j. U sljedećem redu nalazi se cijeli broj q $(0 \le q \le n-3)$ — broj koraka. U j-tom od sljedećih q redova nalazi se prirodan broj k_j $(1 \le k_j \le n)$ koji označava da se u j-tom koraku briše vrh A_{k_j}. Možete pretpostaviti da su u poligonu P vrhovi A_j zadani u smjeru suprotnom od kazaljke na satu, da ne postoje dvije uzastopne paralelne dužine te da su svi indeksi k_j međusobno različiti.

Izlazni podaci

Ispišite $q+1$ red. U prvi red ispišite maksimalnu osvijetljenost početnog poligona P, a u j-ti od sljedećih q redova ispišite maksimalnu osvijetljenost poligona P_j dobivenog nakon j koraka. Za svaki red izlaza, tolerirat će se apsolutno i relativno odstupanje od službenog rješenja za 10^{-5}.

Bodovanje

Podzadatak	Broj bodova	Ograničenja
1	12	$n \le 100$
2	14	$n \le 2000$
3	14	$n \le 100\,000, q = 0$
4	29	$n \le 100\,000$, za svaki $j = 1, \ldots, q-1$ vrijedi $k_j < k_{j+1}$
5	31	$n \le 100\,000$

Primjeri test podataka

ulaz

```
4
0 0
10 0
10 10
0 10
1
2
```

izlaz

```
20.000000
24.142136
```

ulaz

```
6
2 2
4 0
6 0
8 2
8 4
2 4
3
1
4
3
```

izlaz

```
10.828427
11.300563
10.944272
11.656854
```

Zadatak: Zagonetka

Mislav i Marin su na satu kombinatorike učili o permutacijama te su osmislili jednu zanimljivu igru u kojoj igrač mora pogoditi određene permutacije koje zadovoljavaju neke uvjete. *Permutacija* reda n je niz brojeva $p = (p_1, p_2, \ldots, p_n)$ u kojem se svaki broj između 1 i n pojavljuje točno jednom. *Uvjet* je par različitih brojeva (a, b), oba između 1 i n uključivo. Permutacija p *zadovoljava* uvjet (a, b) ako je $p_a < p_b$.

Igra se odvija na sljedeći način. Marin najprije odabere nula ili više uvjeta i jednu permutaciju p reda n koja ih sve zadovoljava. Na početku igre, Marin porukom pošalje Mislavu samo odabranu permutaciju p (uvjeti ostaju tajni). Mislavov je cilj da odredi leksikografski najmanju i leksikografski najveću permutaciju koja zadovoljava sve Marinove uvjete. U svakom koraku igre, Mislav odabere jednu permutaciju q reda n te je porukom pošalje Marinu, a Marin mu otkrije zadovoljava li ta permutacija q sve tajne uvjete.

Ovo je interaktivni zadatak. Napišite program koji će igrati igru umjesto Mislava. Vaš program mora, za zadanu permutaciju p (duljine najviše 100) koja sigurno zadovoljava tajne uvjete, u najviše 5 000 koraka pronaći leksikografski najmanju i leksikografski najveću permutaciju koja zadovoljava sve uvjete.

Interakcija

Prije interakcije vaš program treba sa standardnog ulaza pročitati sljedeće: U prvom redu nalazi se prirodan broj n — veličina svih permutacija. U sljedećem redu nalazi se n različitih prirodnih brojeva $p_1, p_2, \ldots, p_n$ $(1 \le p_j \le n)$ — permutacija p. Možete pretpostaviti da p zadovoljava sve Marinove uvjete.

Nakon toga, vaš program može Marinu slati *upite* pisanjem na standardni izlaz. Svaki upit treba biti ispisan u zasebni red oblika "`query` $q_1\ q_2\ \ldots\ q_n$" gdje su $q_1, q_2, \ldots q_n$ različiti prirodni brojevi između 1 i n uključivo. Nakon svakog ispisanog upita, vaš program mora napraviti *flush* izlaza te sa standardnog ulaza pročitati Marinov *odgovor* — broj 1 ako permutacija q iz upita zadovoljava sve uvjete, a 0 inače.

Kada je vaš program pronašao rješenje, treba na standardni izlaz ispisati red koji sadrži naredbu "`end`", zatim red oblika "$a_1\ a_2\ \ldots\ a_n$" koji sadrži traženu leksikografski najmanju permutaciju te red oblika "$b_1\ b_2\ \ldots\ b_n$" koji sadrži traženu leksikografski najveću permutaciju. Nakon svega, program opet treba napraviti *flush* izlaza i završiti izvođenje.

Napomena: Putem sustava za evaluaciju možete preuzeti primjere koda koji na ispravan način rade interakciju uključujući *flush* izlaza.

Primjer interakcije

U sljedećem primjeru interakcije, prvi stupac sadrži podatke koje vaš program ispisuje na standardni izlaz, a drugi stupac podatke koje vaš program čita sa standardnog ulaza. Nakon tri koraka igre, odnosno nakon tri upita, program određuje točno rješenje.

Izlaz	Ulaz	Napomena
	`4`	odabrani tajni uvjeti su $(2,1)$ i $(3,4)$
	`3 2 1 4`	
`query 2 3 1 4`	`0`	nije zadovoljen uvjet $(2,1)$
`query 3 2 4 1`	`0`	nije zadovoljen uvjet $(3,4)$
`query 4 1 2 3`	`1`	oba uvjeta su zadovoljena
`end`		
`2 1 3 4`		
`4 3 1 2`		

Testiranje

Svoja rješenja možete testirati na dva načina, lokalno ili putem sustava za evaluaciju. Za obje varijante potrebno je najprije izraditi ulaznu datoteku koja će sadržavati test primjer kojim želite testirati svoje rješenje. Ulazna datoteka mora biti oblikovana prema sljedećim pravilima: U prvom redu nalazi se prirodni n — veličina svih permutacija u igri. U sljedećem redu nalazi se n različitih prirodnih brojeva $p_1, p_2, \ldots, p_n$ $(1 \leq p_j \leq n)$ — permutacija p. U sljedećem redu nalazi se cijeli broj m $(0 \leq m \leq 10\,000)$ — broj uvjeta. U svakom od sljedećim m redova nalaze se dva različita prirodna broja a i b $(1 \leq a, b \leq n)$ koja predstavljaju jedan uvjet. Permutacija p mora zadovoljavati svih m uvjeta.

Na primjer, ulaz koji odgovara primjeru interakcije gore je:

```
4
3 2 1 4
2
2 1
3 4
```

Za testiranje putem sustava za evaluaciju potrebno je prvo predati izvorni kod vašeg rješenja putem stranice “Submit”, a zatim poslati test podatak putem stranice “Test”. Lokalno testiranje vrši se putem datoteke “`zagonetka_test`” koju je moguće dohvatiti putem sustava za evaluaciju. Testiranje se vrši sljedećom naredbom:

```
./zagonetka_test ./vase_rjesenje ulazna_datoteka
```

U slučaju testiranja putem sustava za evaluaciju, na izlaz ćete dobiti informaciju je li vaš program točno riješio vaš test primjer, te informacije o naredbama koje je vaš program zadao i odgovorima koje je vaš program dobio.

U slučaju lokalnog testiranja, nećete dobiti informaciju o tome je li program točno riješio test primjer već se sami morate u to uvjeriti. Informacije o tijeku igre će biti zapisane u datoteku `zagonetka.log` u trenutnom direktoriju.

Bodovanje

Podzadatak	Broj bodova	Ograničenja
1	9	$2 \leq n \leq 6$
2	18	$30 \leq n \leq 70$, Marin je odabrao točno 1 uvjet
3	22	$10 \leq n < 30$
4	51	$70 < n \leq 100$