

Školsko natjecanje iz informatike

Srednja škola
Prva podskupina (1. i 2. razred)

24. siječnja 2020.

Zadaci

Ime zadatka	Vremensko ograničenje	Broj bodova
HB	5 sekundi	40
Geo	5 sekundi	50
Statistika	5 sekundi	60
Ukupno		150



Agencija za odgoj i obrazovanje
Education and Teacher Training Agency



HRVATSKI SAVEZ
INFORMATIČARA



Ministarstvo
znanosti i
obrazovanja

Zadatak: HB

U domaćoj glazbenoj praksi, idući su tonovi označeni po jednim slovom:

C, D, E, F, G, A, B, H.

Pritom B možemo označiti i kao A#, što znači A povišeni. Između nekih od ovih tonova postoje međutonovi koje označavamo kao C# (ili Db, što znači D sniženi), D# (ili Eb), F# (ili Gb) i G# (ili Ab).

Ove su oznake u skladu s njemačkom notacijom koja se koristi u još nekim europskim zemljama. Međutim, u međunarodnoj notaciji (primjerice u državama engleskog govornog područja) postoji razlika u oznakama dvaju tonova. Naš ton H odgovara međunarodnom B, dok naš B odgovara međunarodnom A# (ili Bb). Razlika je precizno ilustrirana u donjoj tablici koja za svaki od dvanaest tonova oktave prikazuje englesku i njemačku oznaku (ili dvije moguće oznake) za taj ton.¹

engleski	C	C# Db	D	D# Eb	E	F	F# Gb	G	G# Ab	A	A# Bb	B
njemački/hrvatski	C	C# Db	D	D# Eb	E	F	F# Gb	G	G# Ab	A	A# B	H

Napišite program koji niz tonova pretvara iz "engleske" u "njemačku" notaciju ili obrnuto. Preciznije, program treba prepoznati o kojoj je notaciji riječ te ispisati niz u onoj drugoj notaciji. Ako nije moguće jednoznačno odrediti o kojoj je notaciji riječ, pretpostavite da je riječ o njemačkoj notaciji (koju treba pretvoriti u englesku). U slučaju više mogućih oznaka za isti ton, treba ispisati oznaku s **povisilicom** (znakom #) čak i ako je u ulaznom nizu (na istome mjestu) zadana oznaka sa snizilicom (znakom b).

Ulazni podatci

U prvom je retku neprazan niz od najviše 20 tonova. Niz je ispravan, tj. sadrži tonove koji svi pripadaju jednom od redaka gornje tablice. Tonovi **nisu** odvojeni razmacima!

Izlazni podatci

U jedini redak ispišite traženi niz tonova.

Bodovanje

U testnim primjerima vrijednim 40% bodova pojavljuje se samo **jedan ton**.

Probni primjeri

ulaz

Gb

izlaz

F#

ulaz

ABHC

izlaz

AA#BC

ulaz

DDbD#DBbBAb

izlaz

DC#D#DA#HG#

Pojašnjenje: dajući prednost oznakama s povisilicom, u prvom i drugom primjeru pretvaramo iz njemačke u englesku notaciju, a u trećem iz engleske u njemačku.

¹Za potrebe ovog zadatka držimo se notacija iz tablice, iako su u nekim kontekstima mogući i zapisi poput E# ili Fb ili B#.

Zadatak: Geo

Geohash je metoda koja zemljopisne koordinate pretvara u cijeli broj u svrhu učinkovitog pohranjivanja i upita u bazama podataka. U ovom zadatku promatramo zemljovid kao $2^n \times 2^n$ kvadrat u standardnom koordinatnom sustavu gdje x koordinata raste udesno, a y koordinata raste prema gore. *Ćelija* je jedinični kvadrat poravnat s koordinatnim osima kojemu je donji-lijevi vrh točka s cjelobrojnim koordinatama (x, y) takva da je $0 \leq x, y < 2^n$.

Ukupno je 2^{2n} ćelija na zemljovidu $2^n \times 2^n$. Za neku ćeliju c , njezin geohash $h(c)$ je $2n$ -bitni nenegativni cijeli broj konstruiran bit-po-bit počevši od najznačajnijeg bita, na način da je početni *okvir* jednak cijelom zemljovidu te se sljedeći koraci ponove n puta:

1. Podijelimo okvir na dva jednaka područja – lijevu i desnu polovinu. Ako je ćelija c u lijevoj polovini, sljedeći je bit 0, inače je sljedeći bit 1. Novi okvir je područje koje sadrži ćeliju c .
2. Okvir dijelimo na dva jednaka područja – donju i gornju polovinu. Ako je ćelija c u donjoj polovini, sljedeći bit je 0, inače je sljedeći bit 1. Novi okvir je područje koje sadrži ćeliju c .

Na desnoj slici prikazani su geohashevi (pretvoreni iz binarnog u dekadski zapis) svih ćelija na 8×8 zemljovidu (za $n = 3$) pri čemu se ishodište $(0, 0)$ koordinatnog sustava nalazi u donjem-lijevom vrhu.

21	23	29	31	53	55	61	63
20	22	28	30	52	54	60	62
17	19	25	27	49	51	57	59
16	18	24	26	48	50	56	58
5	7	13	15	37	39	45	47
4	6	12	14	36	38	44	46
1	3	9	11	33	35	41	43
0	2	8	10	32	34	40	42

Na primjer, ćelija s koordinatama donjeg-lijevog vrha u $(4, 2)$ ima geohash $36 = 100100_2$ jer tu ćeliju sužavanjem okvira redom nalazimo u desnom području (1), pa donjem području (0), pa lijevom (0), pa gornjem (1), pa lijevom (0), pa donjem (0).

Napišite program koji za dani geohash $h(c)$ pronalazi koordinate odgovarajuće ćelije c .

Ulazni podatci

U prvom je retku prirodan broj n ($1 \leq n \leq 30$) koji definira dimenziju zemljovida $2^n \times 2^n$.

U sljedećem je retku cjelobrojni geohash h ($0 \leq h < 2^{2n}$).

Izlazni podatci

U prvi redak ispišite tražene koordinate x i y donjeg-lijevog vrha tražene ćelije, odvojene razmakom.

Bodovanje

U testnim primjerima vrijednim 50% bodova vrijedi $n \leq 8$.

Probni primjeri

ulaz	ulaz	ulaz
1	2	3
0	7	36
izlaz	izlaz	izlaz
0 0	1 3	4 2

Zadatak: Statistika

Na Mirkovu Linux računalu, svaka datoteka ima hijerarhijsku putanju, primjerice:

```
/core.dat  
/docs/biljeske.txt  
/skolsko/zadatci/geo/primjeri/primjer.in  
/skolsko/upute.pdf
```

Tako se npr. treća navedena datoteka `primjer.in` nalazi unutar direktorija `primjeri`, koji se nalazi unutar direktorija `geo`, koji se nalazi unutar direktorija `zadatci`, koji se nalazi unutar direktorija `skolsko`, koji se nalazi u početnom (tzv. *root*) direktoriju.

Svaka datoteka ima svoju *ekstenziju* – završetak imena datoteke, npr. `.html`, `.cpp`, `.py`, `.zip`... U gornjem primjeru ekstenzije su `.dat`, `.txt`, `.in` i `.pdf`. Također, svaka datoteka ima svoju *razinu*, koja se definira kao broj poddirektorija u koje je potrebno ući da bi se iz početnog direktorija došlo do određene datoteke. Tako se prva datoteka iz gornjeg primjera nalazi na razini 0 (jer se već nalazi u početnom direktoriju), druga na razini 1 (jer pripada direktoriju `docs`), treća na razini 4, a četvrta na razini 1.

Mirko želi napraviti statistiku svojih datoteka. Odlučio je zasebno promatrati svaku razinu i za sve datoteke koje pripadaju istoj razini, Mirko želi za svaku ekstenziju izračunati **ukupnu veličinu datoteka te ekstenzije**. Pritom ekstenzije (za pojedinu razinu) želi poredati silazno po broju datoteka koje imaju tu ekstenziju, potom (ako ima izjednačenja) silazno prema ukupnoj veličini datoteka ekstenzije, a potom (ako još uvijek ima izjednačenja) abecedno po ekstenziji.

Primjerice, na razini 1 u gornjem primjeru nalaze se ekstenzije `.txt` i `.pdf`. Budući da su prema prvom kriteriju (broj datoteka) one izjednačene (po jedna datoteka svake ekstenzije), prije će doći ona ekstenzija čija je ukupna veličina datoteka (u kilobajtima) veća.

Napišite program koji pomaže Mirku i ispisuje traženu statistiku zadanog skupa njegovih datoteka.

Ulazni podatci

U prvom je retku prirodan broj n ($1 \leq n \leq 30$), broj promatranih datoteka. U svakom od sljedećih n redaka navedena je putanja datoteke i njezina veličina u kilobajtima, odvojene razmakom. Veličina je prirodan broj manji od 10^7 .

Putanja datoteke sastoji se od najviše 50 znakova, bez razmaka. Putanja počinje znakom `/` nakon kojeg slijedi niz od nula ili više direktorija odvojenih istim znakom `/`, nakon čega slijedi ime datoteke i njezina ekstenzija odvojena točkom. Ime svakog direktorija, datoteke i ekstenzije neprazan je niz malih slova engleske abecede. Nijedna putanja neće biti identična nekoj drugoj.

Izlazni podatci

Redom za svaku razinu $r = 0, 1, 2, \dots$, ako se na njoj pojavljuje barem jedna datoteka, treba ispisati blok redaka. U prvom retku bloka treba stajati sama razina r . U svakom od sljedećih E_r redaka bloka (gdje je E_r broj različitih ekstenzija koje se javljaju na razini r) treba ispisati odgovarajuću ekstenziju (uključujući vodeću točku) i ukupnu veličinu u kilobajtima svih datoteka te ekstenzije na razini r , odvojene razmakom. Ekstenzije unutar bloka za razinu r treba poredati u skladu s kriterijima iz teksta zadatka (broj datoteka, veličina, abecedno).

Probni primjeri

ulaz

```
4
/core.dat 100
/tmp.dat 150
/install.zip 300
/tmp/install.zip 400
```

izlaz

```
0
.dat 250
.zip 300
1
.zip 400
```

ulaz

```
8
/test/in.x 100
/test/in.b 30
/test/out.x 100
/test/out.b 30
/test/pic/in.svg 10
/test/pic/out.svg 10
/x/test/in.x 20
/x/test/in.b 20
```

izlaz

```
1
.x 200
.b 60
2
.svg 20
.b 20
.x 20
```