

Županijsko natjecanje iz informatike

Srednja škola
Druga podskupina (3. i 4. razred)

10. veljače 2017.

Zadaci

Ime zadatka	Vremensko ograničenje	Memorijsko ograničenje	Broj bodova
Ukulele	1 sekunda	512 MiB	30
Turnir	1 sekunda	512 MiB	40
Doseg	1 sekunda	512 MiB	60
Led	10 sekundi	512 MiB	70
Ukupno			200

Zadatak: Ukulele

Tablatura je način zapisivanja glazbe koji opisuje redosljed i način trzanja žica na instrumentu. Pretpostavimo da opisujemo glazbeno djelo duljine n taktova (takt je jedinična mjera vremena) označenih brojevima od 1 do n . Djelo je napisano za *ukulele* — instrument sličan gitari koji ima četiri žice označene brojevima od 1 do 4. Tablatura glazbenoga djela je tablica znakova koje se sastoji od 4 retka i n stupaca — ukoliko je u i -tom taktu potrebno trznuti j -tu žicu onda je i -ti znak u j -tom retku tablature jednak “0” (znamenka nula), a inače je “-” (minus). Prilikom zapisa tablature, prije prvog i nakon zadnjeg stupca dodajemo granični stupac sastavljen od znakova “|” (vertikalna crta, ASCII 124).

Za zadanu tablaturu, odredite oznake i redosljed žica koje je potrebno trznuti kako bi odsvirali glazbeno djelo. U ovom zadatku razmatramo samo jednostavna djela pa je u svakom taktu potrebno trznuti najviše jednu žicu.

Ulazni podaci

Ulaz se sastoji od četiri reda jednake duljine. U j -tom redu se nalazi j -ti redak tablature zajedno s graničnim znakovima. Prvi i zadnji znak svakog retka je “|”, dok je svaki ostali znak ili “0” ili “-”. Djelo se sastoji od najviše 100 taktova ($n \leq 100$), što znači da je svaki red ulaza duljine najviše 102 znaka.

Možete pretpostaviti da se u svakom stupcu pojavljuje najviše jedan znak “0” te da se barem jedan znak “0” pojavljuje na ulazu. Dopušteno je da se u nekim stupcima ne pojavljuje niti jedan znak “0”.

Izlazni podaci

Ispišite m redova gdje je m ukupan broj trzanja žice (ukupan broj znakova “0” na ulazu). U k -ti red ispišite oznaku k -te po redu žice koje je potrebno trznuti prilikom sviranja djela.

Primjeri test podataka

ulaz

```
|0---|  
|-0--|  
|--0-|  
|---0|
```

izlaz

```
1  
2  
3  
4
```

ulaz

```
|-0---0----|  
|--0-0-----|  
|-----0--|  
|-----0|
```

izlaz

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

ulaz

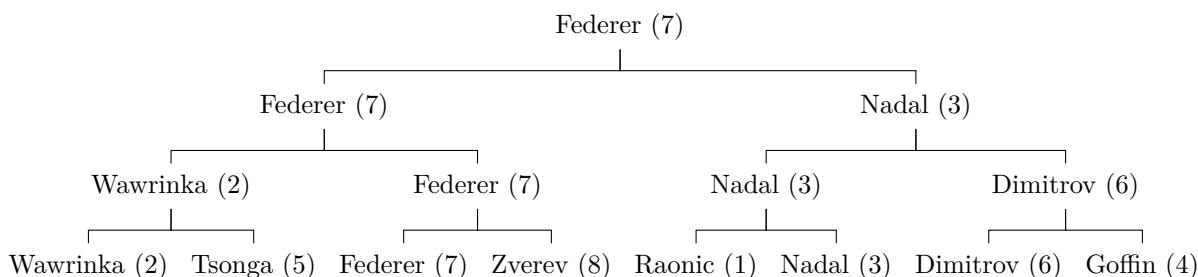
```
|-----000--|  
|-----|  
|----0-0-----|  
|--0-----|
```

izlaz

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

Zadatak: Turnir

Mirko je veliki ljubitelj tenisa te prije svakog turnira pokušava predvidjeti rasplet svih mečeva od prvog kola pa sve do finala. Svoja predviđanja zapisuje u strukturiranu tablicu koju nazivamo *stablo turnira*. Ako se turnir sastoji od n kola, onda na njemu sudjeluje 2^n igrača označenih redom brojevima od jedan na dalje te se stablo turnira sastoji od $n + 1$ redaka. U najdonji redak zapisujemo jedan za drugim, slijeva na desno, parove igrača koji međusobno igraju u prvom kolu. Dakle, u prvom kolu međusobno igraju prvi i drugi po redu igrač iz zadnjeg retka, treći i četvrti, i tako dalje. U redak iznad zapisujemo redom pobjednike tih mečeva, a parove sljedećeg kola opet čine prvi i drugi igrač u retku, treći i četvrti i tako dalje. Postupak ponavljamo sve dok ne dođemo do prvog retka u kojem je zapisan pobjednik turnira.



Slika 1: Originalno stablo turnira iz drugog primjera test podataka

Mirko je iz novima u najdonji redak tablice prepisao ždrijeb prvog kola turnira te prema svojim predviđanjima popunio ostatak stabla naljepnicama s imenima tenisača. Njegova sestra je, pod okriljem noći, odlijepila sve naljepnice, dobro ih promiješala, i zalijepila nazad na stablo. Mirko je shvatio da nešto ne valja, ali je nakon početnog šoka odahnuo jer je shvatio da će ipak moći rekonstruirati njegova predviđanja.

Zadano je stablo turnira u kojem zadnji redak odgovara originalnom, dok su ostali elementi možda izmiješani. Odredite originalno stablo turnira. Možete pretpostaviti da rješenje postoji te da je jedinstveno.

Ulazni podaci

Prvi red sadrži prirodni broj n ($1 \leq n \leq 10$) — broj kola u turniru. U k -tom od sljedećih $n + 1$ redova nalazi se 2^{k-1} prirodnih brojeva — oznake igrača u k -tom redu stabla turnira slijeva nadesno. Svaka oznaka igrača je prirodni broj između 1 i 2^n uključivo. Sve oznake u zadnjem redu su različite.

Izlazni podaci

Ispišite $n + 1$ redova koji sadržavaju originalno stablo turnira u istom obliku kao na ulazu. Ispisano stablo mora biti ispravno stablo turnira od kojeg se može dobiti stablo na ulazu postupkom opisanim u tekstu zadatka. Između ostalog, zadnji red izlaza mora biti potpuno jednak zadnjem redu ulaza.

Primjeri test podataka

ulaz

1
1
2 1

izlaz

1
2 1

ulaz

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

izlaz

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

ulaz

2
3
2 2
4 3 2 1

izlaz

2
3 2
4 3 2 1

Zadatak: Doseg

Razmatramo pravila deklaracije i dosega varijabli u jednostavnom programskom jeziku sintaksom sličnom Pascal-u. Program u ovom jeziku se sastoji od linija, gdje svaka linija sadrži jednu od sljedećih naredbi:

- “`var c = k`”, gdje je c malo slovo engleske abecede, a k prirodni broj: ova naredbu zovemo *deklaracija varijable* c . Osim što deklarira varijablu c , ova naredba i postavlja njenu vrijednost na broj k .
- “`write c`”: ova naredba ispisuje vrijednost varijable c .
- “`begin`”: početak bloka naredbi.
- “`end`”: završetak bloka naredbi.

Blok naredbi započinje naredbom “`begin`”, sadrži nula, jednu ili više linija te uvijek završava naredbom “`end`”. Blokovi također mogu sadržavati druge blokove, koji opet mogu sadržavati druge blokove itd. *Tijelo* bloka A je skup svih naredbi koje se nalaze *direktno* unutar tog bloka (dakle nisu uključene naredbe koje se nalaze unutar nekog bloka B unutar A).

Neka je D deklaracija “`var c = k`” u tijelu bloka A . *Doseg* deklaracije D je skup svih naredbi koje se nalaze (direktno ili indirektno) unutar bloka A , a u programu dolaze nakon D . U našem primjeru, doseg deklaracije “`var b = 4`” iz linije 3 programa je skup svih naredbi između linija 4 i 19, dok je doseg deklaracije “`var b = 6`” iz linije 8 samo naredba u liniji 9.

Neka je C naredba “`write c`”. Varijabla c u ovoj naredbi je *deklarirana* naredbom D ako se C nalazi u dosegu od D (te naravno D deklarira upravo varijablu c). Ukoliko postoji više takvih deklaracija D , onda prioritet ima ona koja se pojavljuje *najkasnije* u programu. Tako je, na primjer, varijabla “ b ” u liniji 9 deklarirana u liniji 8, dok je ista varijabla u liniji 11 deklarirana u liniji 3. Naredba “`write c`” ispisuje vrijednost koja je pridružena varijabli c u njenoj deklaraciji.

Program je jedan blok naredbi. Za zadani program odredite izlaz svake od njegovih “`write`” naredbi. Možete pretpostaviti da je program ispravan te da je svaka varijabla deklarirana u trenutku korištenja. Primijetite da je dopušteno deklarirati istu varijablu više puta unutar tijela jednog bloka.

```
1 begin
2   var a = 3
3   var b = 4
4   begin
5     write a
6     var a = 5
7     write a
8     var b = 6
9     write b
10  end
11  write b
12  begin
13    write b
14    begin
15      var a = 10
16      write a
17    end
18    write a
19  end
20 end
```

Ulazni podaci

U prvom redu nalazi se prirodni broj n ($n \leq 1000$) — broj linija u programu. U j -tom od sljedećih n redova nalazi se j -ta linija programa. Svaka linija sadrži jednu od četiri naredbe iz teksta zadatka. Oznaka varijable c u naredbama je uvijek malo slovo engleske abecede, dok je k prirodni broj manji od 1000. Prije oznake varijable c , broja k te znaka “`=`” dolazi točno jedan razmak. Možete pretpostaviti da je program ispravan prema pravilima iz teksta zadatka.

Izlazni podaci

Ispišite m redova gdje je m ukupan broj “`write`” naredbi u programu. U j -ti red ispišite prirodni broj koji je bi ispisala j -ta po redu “`write`” naredba u programu.

Bodovanje

- U test podacima vrijednim 30% bodova program sadrži točno jedan blok (dakle prva naredba programa je “`begin`”, zadnja je “`end`”, a svaka preostala naredba je ili “`var`” ili “`write`”).

Primjeri test podataka

ulaz

```
10
begin
var a = 4
var x = 6
var z = 1
write a
write z
write x
var a = 5
write a
end
```

izlaz

```
4
1
6
5
```

ulaz

```
20
begin
var a = 3
var b = 4
begin
write a
var a = 5
write a
var b = 6
write b
end
write b
begin
write b
begin
var a = 10
write a
end
write a
end
end
```

izlaz

```
3
5
6
4
4
10
3
```

ulaz

```
21
begin
begin
var t = 5
write t
var a = 1
end
begin
var t = 4
write t
begin
var t = 3
end
write t
end
begin
begin
end
begin
end
end
end
```

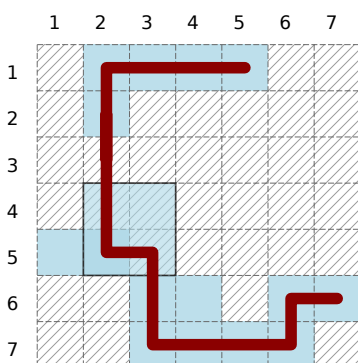
izlaz

```
5
4
4
```

Zadatak: Led

Ledolomac se na djelomično smrznutom Dunavu mora probiti od točke A do točke B . Mapa rijeke je kvadratna mreža znakova koja se sastoji od n redaka označenih brojevima od 1 do n odozgo prema dolje, te n stupaca označenih brojevima od 1 do n slijeva nadesno. Neka polja u mapi su zaleđena (označena znamenkom "1") dok ostala nisu (označena znamenkom "0"). Ledolomac se može micati u četiri smjera (gore, dolje, lijevo ili desno) te se može kretati i preko slobodnih i preko zaleđenih polja. Međutim, kretanje po zaleđenim poljima je puno teže pa posada želi naći put od A do B koji sadrži što manje takvih polja.

Dodatno, posada ima na raspolaganju jednu kemijsku bombu koju mogu precizno lansirati bilo gdje na rijeku. Pomoću bombe *dosega* k je moguće otopiti sva zaleđena polja unutar jednog kvadratnoga područja veličine k redaka puta k stupaca (koje mora u potpunosti ležati unutar mape).



Slika 2: Rješenje jedinog scenarija iz prvog primjera test podataka

Zadana je mapa rijeke te q različitih scenarija. U svakom scenariju j su zadana polja A_j i B_j te doseg bombe k_j . Za svaki scenarij odredite najmanji mogući broj zaleđenih polja koje ledolomac treba prijeći na zadanoj mapi kako bi se probio od A_j do B_j , uz upotrebu jedne bombe dosega k_j .

Ulazni podaci

U prvom redu nalazi se prirodni broj n ($1 \leq n \leq 2000$) — dimenzije mape. U j -tom od sljedećih n redova nalazi se niz od točno n znakova "0" ili "1" — j -ti redak mape.

U sljedećem redu nalazi se prirodni broj q ($1 \leq q \leq 5$) — broj scenarija. U j -tom od sljedećih q redova nalazi se pet prirodnih brojeva $k, r_{A_j}, s_{A_j}, r_{B_j}$ i s_{B_j} ($1 \leq k_j \leq n, 1 \leq r_{A_j}, s_{A_j}, r_{B_j}, s_{B_j} \leq n$) — doseg bombe te koordinate (broj retka pa broj stupca) polja A_j i polja B_j u j -tom scenariju. U svakom scenariju su polja A_j i B_j međusobno različita te niti jedno od njih nije zaleđeno.

Izlazni podaci

Ispišite q redova. U j -ti red ispišite traženi minimalni broj zaleđenih polja u j -tom scenariju.

Bodovanje

- U test podacima vrijednim 20% bodova vrijedi $n \leq 20$
- U dodatnim test podacima vrijednim 20% bodova vrijedi $n \leq 200$.

Primjeri test podataka

ulaz

7
1000011
1011111
1111111
1111111
0011111
1100100
1100001
1
2 1 5 6 7

izlaz

1

ulaz

8
01111100
10111101
11101110
11010110
11000100
11010111
11101101
01100010
4
2 1 1 8 8
2 1 8 8 1
3 1 1 8 8
5 1 8 8 1

izlaz

3
2
2
1

Pojašnjenje prvog primjera: Ukoliko posada lansira bombu tako da se otopi sav led u kvadratu čiji je gornji lijevi kut polje (4,2) onda postoji put od A do B koji prolazi kroz samo jedno zaleđeno polje.