

## PSA2: DN3

Šov na televiziji se nadaljuje, saj se bliža Evrovizija, a zapletlo se je že na začetku! Po pregledu  $P$  pesmi, ki se bodo letos borile za zmago, je komisija odkrila še  $D$  demoposnetkov iz prejšnjih let, ki so sumljivo podobni izsekom pesmi kandidatk. Pogoji za nastop na Evroviziji je seveda ta, da je pesem nova, zato pomagajmo komisiji in za vsak demoposnetek identificirajmo

- kateri pesmi je podoben,
- kateremu delu izbrane pesmi je podoben.

Pesmi (in demoposnetki) so letos tako preproste, da jih lahko predstavimo kot zaporedje frekvenc tonov, kar pomeni, da je v vsakem trenutku v pesmi prisotna le ena frekvenca. Pesmi trajajo  $p$  sekund, demoposnetki pa  $d$  sekund. Frekvenco zajamemo 100-krat na sekundo, zato je dolžina seznamov frekvenc, ki predstavljajo pesmi oz. demoposnetke, enaka  $\ell_p = 100p$  oz.  $\ell_d = 100d$ .

### Naloga A (5 točk)

Goljufivci niso bili ravno previdni in originalne pesmi niso nič priredili, zato se demo v pesmi pojavi kot podseznam, tj. za vsak demo  $a$  lahko najdemo tako pesem  $b$  in indeks  $i$ , da velja  $b[i : i + \ell_d] = a$ .

### Naloga B (5 točk)

Goljufivci so bili nekoliko previdnejši in so svojo pesem posneli v drugi tonaliteti. Človeško uho sliši to kot *isto pesem*, ki pa je zaigrana višje ali nižje (tukaj je dostopna Tetrisova pesem v različnih tonalitetah). V jeziku frekvenc to pomeni, da se vsaka frekvenca  $\nu$  pomnoži z nekim faktorjem  $q$ , ki je odvisen od tega, kako močno smo spremenili tonaliteto<sup>1</sup>. To pomeni, da sedaj za dani demo  $a$  lahko najdemo tako pesem  $b$  in indeks  $i$ , da velja  $b[i : i + \ell_d] = [q \cdot a[0], q \cdot a[1], \dots, q \cdot a[\ell_d - 1]]$ .

### Vhodni podatki

Vsaki od podnalog pripada vhodna datoteka (**nalogaA.txt** in **nalogaB.txt**). V prvi vrstici vhodne datoteke se nahajajo s presledki ločena števila  $P$ ,  $D$ ,

---

<sup>1</sup>Velja:  $q = 2^{\Delta/12}$ , kjer je  $\Delta \in \mathbb{Z}$  število poltonov, za katero zamaknemo pesem.

$\ell_p$  in  $\ell_d$ , kjer je  $P$  število pesmi,  $D$  število demoposnetkov,  $\ell_p$  dolžina seznamov, ki predstavljajo pesmi, in  $\ell_d$  dolžina seznamov, ki predstavljajo demoposnetke.

Tej vrstici sledi  $P$  vrstic, v katerih so podane pesmi. Vsaka pesem je podana kot seznam  $\ell_p$  realnih števil, ločenih s presledki. Za pesmimi pride  $D$  vrstic, ki podajo demoposnetke na analogen način.

Vse frekvence so bile izračunane natančno, kolikor se da, nato pa podane na dve decimalni mesti natančno (npr. 440.00).

## Izhod

Naj bo  $\mathcal{P}$  seznam vseh pesmi. Za vsak demoposnetek v izhodno datoteko zapišite s presledkom ločena indeksa  $i$  in  $z$ , za katera velja, da je  $\mathcal{P}[i][z : z + \ell_d]$  enak (morda frekvenčno zamaknjenemu) demoposnetku.

## Preverjanje rešitev

Definirajte, kolikor razredov želite, nato pa pošinite svojo kodo v priloženem razredu `Main`. Rešitev se bo nato avtomatsko preverila s pomočjo razreda `PreveriResitev`. Za preverjanje rešitev potrebujete svojo izhodno datoteko in uradne rešitve v datotekah `nalogaA.sol` in `nalogaB.sol`. Rešitev za dano nalogo je pravilna, če

- je dovolj hitra (za vsako od nalog A in B imate 15 sekund) in
- so vsi odgovori pravilni.

Končno verzijo kode oddajte na učilnici (samo izvorno kodo), ker jo bomo izvajalci predmeta še enkrat preverili.

## Namig za reševanje

Morda bo treba kje izračunati korelacijo. Premisli, kako je ta povezana s skalarnim produktom. V pomoč pri hitrejšem računanju pa naj bo še naslednje. Če imamo seznam  $s$  dolžine  $m$ , lahko izračunamo povprečja vseh podseznamov dolžine  $n$  v skupnem času  $\mathcal{O}(m)$ , tako da najprej izračunamo vsoto  $v = \sum_{i=0}^{n-1} s[i]$ , nato pa ponavljamo spodnji postopek in na  $j$ -tem koraku

1. izračunamo povprečje  $v/n$
2. posodobimo vsoto  $v$  na  $v = v + v[j+n] - v[j]$ .

Podoben postopek lahko izpeljemo za računanje variance, če jo računamo po formuli  $\text{Var}(X) = \mathbb{E}(X^2) - \mathbb{E}^2(X)$  in hranimo vsoto ter vsoto kvadratov.

## Primer

Vhod:

```
4 3 10 5
1.10 2.20 3.30 1.10 2.20 4.40 1.10 2.20 5.50 1.10
1.10 0.00 2.20 0.00 3.30 0.00 4.40 0.00 5.50 0.00
1.10 2.20 3.30 4.40 1.10 2.20 3.30 4.40 5.50 0.00
9.90 8.80 7.70 6.60 9.90 8.80 7.70 6.60 5.50 0.00
3.30 1.10 2.20 4.40 1.10
1.10 2.20 3.30 4.40 5.50
1.10 2.20 4.40 1.10 2.20
```

Izhod:

```
0 2
2 4
0 3
```

Vhod:

```
4 3 10 5
1.10 2.20 3.30 1.10 2.20 4.40 1.10 2.20 5.50 1.10
1.10 0.00 2.20 0.00 3.30 0.00 4.40 0.00 5.50 0.00
1.10 2.20 3.30 4.40 1.10 2.20 3.30 4.40 5.50 0.00
9.90 8.80 7.70 6.60 9.90 8.80 7.70 6.60 5.50 0.00
6.60 2.20 4.40 8.80 2.20
2.20 4.40 6.60 8.80 11.00
0.55 1.10 2.20 0.55 1.10
```

Izhod:

```
0 2
2 4
0 3
```

Vidimo, da si je morda več pesmi izposodilo isti demoposnetek. Drugi vhod je mogoč le pri nalogi B (faktorji  $q$  so po vrsti  $1/2$ ,  $1/2$  in  $2$ ).