

# Programiranje 1 — sedma domača naloga

Rok za oddajo: nedelja, 15. decembra 2019, ob 23:55

Do-re-mi-...

## Uvod

Glasbena *nota* predstavlja ton določene frekvence. Note zapisujemo kot *oznaka<sub>oktava</sub>*, pri čemer je *oznaka* besedilna oznaka note (C, Cis, D, Dis, E, F, Fis, G, Gis, A, B, H), *oktava* pa je ne-negativno celo število. Po naraščajočih frekvencah so note urejene takole:

C<sub>0</sub>, Cis<sub>0</sub>, D<sub>0</sub>, Dis<sub>0</sub>, E<sub>0</sub>, F<sub>0</sub>, Fis<sub>0</sub>, G<sub>0</sub>, Gis<sub>0</sub>, A<sub>0</sub>, B<sub>0</sub>, H<sub>0</sub>,  
C<sub>1</sub>, Cis<sub>1</sub>, D<sub>1</sub>, Dis<sub>1</sub>, E<sub>1</sub>, F<sub>1</sub>, Fis<sub>1</sub>, G<sub>1</sub>, Gis<sub>1</sub>, A<sub>1</sub>, B<sub>1</sub>, H<sub>1</sub>,  
C<sub>2</sub>, Cis<sub>2</sub>, D<sub>2</sub>, Dis<sub>2</sub>, E<sub>2</sub>, F<sub>2</sub>, Fis<sub>2</sub>, G<sub>2</sub>, Gis<sub>2</sub>, A<sub>2</sub>, B<sub>2</sub>, H<sub>2</sub>,  
...

Glasbena *lestvica* je navzgor neomejeno zaporedje enakomerno razmaknjenih not.<sup>1</sup> Lestvico določimo z *začetno noto* in *razmakom* med sosednjima notama. Razmak je pozitivno celo število. Na primer, lestvica z začetno noto Fis<sub>4</sub> in razmakom 3 (na kratko jo bomo zapisali kot (Fis<sub>4</sub>, 3)) vsebuje sledeče note:

Fis<sub>4</sub>, A<sub>4</sub>, C<sub>5</sub>, Dis<sub>5</sub>, Fis<sub>5</sub>, A<sub>5</sub>, C<sub>6</sub>, Dis<sub>6</sub>, Fis<sub>6</sub>, A<sub>6</sub>, C<sub>7</sub>, Dis<sub>7</sub>, Fis<sub>7</sub>, ...

Lestvica (Fis<sub>4</sub>, 5) pa vsebuje sledeče note:

Fis<sub>4</sub>, H<sub>4</sub>, E<sub>5</sub>, A<sub>5</sub>, D<sub>6</sub>, G<sub>6</sub>, C<sub>7</sub>, F<sub>7</sub>, B<sub>7</sub>, Dis<sub>8</sub>, Gis<sub>8</sub>, Cis<sub>9</sub>, Fis<sub>9</sub>, ...

## Vaša naloga

Napišite razreda **Nota** in **Lestvica** tako, da bodo njuni objekti predstavljali posamezne glasbene note oz. lestvice. Pri vsakem razredu bomo navedli seznam javno dostopnih konstruktorjev in metod, ki jih mora vsebovati. Atributov in privatnih konstruktorjev oz. metod ne predpisujemo; definirajte jih po lastni želji oz. potrebi.

Vaša razreda bomo preizkusili s testnimi razredi in pripadajočimi izhodi (izpisi). V sledečih seznamih konstruktorjev in metod oznaka v oglatih oklepajih pove, kateri skriti testni razredi lahko kličejo ta konstruktor oz. metodo. Na primer, oznaka [31–] pove, da lahko konstruktor oz. metodo kličejo skriti testni razredi 31–50. V testnih razredih 1–30 se klic te metode ne bo pojavil, zato jo lahko mirno izpustite, če ciljate na največ 60% točk.

## Razred Nota

Razred **Nota** naj vsebuje sledeče javno dostopne konstruktorje in metode:

---

<sup>1</sup>V praksi note niso nujno enako razmaknjene, a za potrebe te naloge si bomo življenje nekoliko poenostavili.

- [1–] `public Nota(int oktava, int indeks):`

Ustvari objekt, ki predstavlja noto s podano oktavo in indeksom znotraj oktave (nota C ima indeks 0, nota Cis indeks 1, ..., nota H pa indeks 11). V vseh testnih razredih je parameter `oktava` celo število z intervala  $[0, 10^3]$ , parameter `indeks` pa celo število z intervala  $[0, 11]$ .

Na primer, če je parameter `oktava` enak 4, parameter `indeks` pa 6, potem ustvarimo objekt, ki predstavlja noto Fis<sub>4</sub>.

- [1–] `public int vrniOktavo():`

Vrne oktavo note `this`. Na primer, pri noti Fis<sub>4</sub> je oktava enaka 4.

- [1–] `public int vrniIndeks():`

Vrne indeks note `this` znotraj oktave. Na primer, pri noti Fis<sub>4</sub> je indeks note znotraj oktave enak 6.

- [21–] `public String toString():`

Vrne zapis note `this` v obliki *oznaka\_oktava*. Na primer, za noto Fis<sub>4</sub> naj metoda vrne niz Fis\_4.

- [26–] `public int razmak(Nota nota):`

Vrne razmak med noto `this` in noto `nota`. Na primer, razmak med notama B<sub>3</sub> in Dis<sub>4</sub> znaša −5 (če objekt `this` predstavlja noto B<sub>3</sub>) oziroma 5 (če objekt `this` predstavlja noto Dis<sub>4</sub>).

- [26–] `public boolean jeNizjaKot(Nota nota):`

Vrne vrednost `true` natanko v primeru, če ima nota `this` nižjo frekvenco od note `nota`. Na primer, nota B<sub>3</sub> ima nižjo frekvenco od note Dis<sub>4</sub>.

- [31–] `public Nota naRazdalji(int razmak):`

Vrne noto, ki je za podani razmak oddaljena od note `this`. Na primer, če objekt `this` predstavlja noto B<sub>3</sub>, parameter `razmak` pa je enak 5, bi metoda morala vrniti noto Dis<sub>4</sub>, v primeru razmaka −5 pa bi dobili noto F<sub>3</sub>. Če bi ciljna nota morala imeti negativno oktavo, naj metoda vrne `null`. Parameter `razmak` je v vseh testnih primerih celo število z intervala  $[-10^3, 10^3]$ .

## Razred Lestvica

Razred Lestvica naj vsebuje sledeče javno dostopne konstruktorje in metode:

- [16–] `public Lestvica(Nota zacetnaNota, int razmak):`

Ustvari objekt, ki predstavlja lestvico s podano začetno noto in podanim razmakom. Parameter `razmak` je celo število z intervala  $[1, 10^3]$ .

- [16–] `public Nota vrniZacetnoNoto():`

Vrne začetno noto lestvice `this`.

- [16–] `public int vrniRazmak():`

Vrne razmak lestvice `this`.

- [21–] `public String toString():`

Vrne niz oblike  $(nota, razmak)$ , kjer je *nota* niz, ki ga za začetno noto lestvice `this` vrne metoda `toString`. Na primer, za lestvico z začetno noto  $Fis_4$  in razmakom 3 naj metoda vrne niz  $(Fis_4, 3)$ .

- [26–] `public static Lestvica izPrvihDveh(Nota zacetna, Nota druga):`

Vrne lestvico, ki se prične z noto *zacetna* in nadaljuje z noto *druga*. Nota *druga* ima višjo frekvenco od note *zacetna*.

- [31–] `public Nota clen(int indeks):`

Vrne člen lestvice `this` s podanim indeksom, torej noto, ki je v lestvici za *indeks* mest oddaljena od začetne note. Na primer, za lestvico  $(Fis_4, 5)$  bi moral klic metode `clen` s parametrom 0 vrniti noto  $Fis_4$ , klic s parametrom 3 pa noto  $A_5$ . Parameter *indeks* je celo število z intervala  $[0, 10^3]$ .

- [36–] `public boolean vsebujeNoto(Nota nota):`

Vrne vrednost `true` natanko v primeru, če lestvica `this` vsebuje podano noto.

- [41–] `public int perioda():`

Vrne število različnih oznak not, ki nastopajo v lestvici `this`. Na primer, v lestvici  $(Fis_4, 3)$  imamo 4 različne oznake ( $Fis, A, C, Dis$ ), v lestvici  $(Fis_4, 5)$  pa je različnih oznak 12 (torej vse možne).

- [46–] `public Lestvica presek(Lestvica lestvica):`

Vrne lestvico, ki vsebuje vse skupne note lestvic `this` in *lestvica*. Če lestvici nimata skupnih not, naj metoda vrne `null`.

Na primer, presek lestvic  $(Fis_4, 3)$  in  $(Fis_4, 5)$  je lestvica  $(Fis_4, 15)$ , presek lestvic  $(Fis_4, 5)$  in  $(F_4, 7)$  je lestvica  $(D_6, 35)$ , lestvici  $(Fis_4, 6)$  in  $(F_4, 8)$  pa nimata preseka.

Namig 1: Če se lestvici pričneta z isto noto, naloga ni posebej težka. V nasprotnem primeru pa najprej preverite, ali imata lestvici skupen presek. Če ga imata, poiščite prvo noto, ki je skupna obema lestvicama, in postopajte tako kot v enostavnem posebnem primeru.

Namig 2: Enačba  $ax - by = c$ , kjer so  $a, b$  in  $c$  cela števila, ima rešitev v celoštevilskih  $x$  in  $y$  natanko tedaj, ko je število  $|c|$  večkratnik števila  $\gcd(|a|, |b|)$ , kjer  $\gcd$  označuje največji skupni delitelj podanih števil.

## Oddaja naloge

Oddajte datoteki z nazivoma `Nota.java` in `Lestvica.java`. V prvi vrstici obeh datotek v komentarju navedite vašo vpisno številko. Če je, denimo, vaša vpisna številka enaka 63190999, morata datoteki izgledati takole:

```
// 63190999

public class Nota {
    ...
}
```

```
// 63190999  
  
public class Lestvica {  
    ...  
}
```

Oddajte **ločeni** datoteki `Nota.java` in `Lestvica.java`. Ne »zipajte«!