

# Programiranje 1 — tekmovanje

Rok za oddajo: nedelja, 21. januarja 2024

## Wordle

Vabimo vas k sodelovanju na tekmovanju strojev<sup>1</sup> za igro Wordle. Če se želite pomeriti s svojim strojem, ga oddajte na Učilnico do vključno nedelje, 21. januarja, do 23:59.

### Pravila igre

Pri igri Wordle si eden od igralcev, recimo mu  $A$ , izbere besedo fiksne dolžine  $n$  iz fiksne slovarja, drugi igralec, recimo mu  $B$ , pa jo poskuša uganiti. Igralec  $B$  v vsakem poskusu tvori besedo dolžine  $n$  (ni nujno, da pripada slovarju), igralec  $A$  pa mu odgovori z zaporedjem  $n$  znakov iz množice  $\{+, -, o\}$ , ki pove, katere črke besede, ki jo je tvoril igralec  $B$ , so na pravem mestu (znaki  $+$ ), katere na napačnem (znaki  $o$ ), katere pa v besedi, ki si jo je izbral igralec  $A$ , sploh ne obstajajo (znaki  $-$ ).

Bodimo natančnejši. Recimo, da si je igralec  $A$  izbral besedo  $a = a_0a_1 \dots a_{n-1}$ , igralec  $B$  pa je poskusil z besedo  $b = b_0b_1 \dots b_{n-1}$ . Igralec  $A$  določi svoj odgovor (zaporedje  $z = z_0z_1 \dots z_{n-1}$ ) takole:

- Za vsak  $i \in \{0, \dots, n-1\}$ , pri katerem je  $a_i = b_i$ , nastavi  $z_i$  na  $+$ . Vse pripadajoče črke besed  $a$  in  $b$  prečrta, da jih izloči iz nadaljnje obravnave.
- Če obstaja  $i \in \{0, \dots, n-1\}$ , tako da črka  $b_i$  obstaja v besedi  $a$  (recimo, da je  $b_i = a_j$  za nek  $j \neq i$ ), potem nastavi  $z_i$  na  $o$ , takoj zatem pa črki  $b_i$  in  $a_j$  prečrta. Postopek ponovi za vse take indekse  $i$ .
- Za vse ostale  $i \in \{0, \dots, n-1\}$  nastavi  $z_i$  na  $-$ .

Recimo, da igralec  $A$  izbere besedo **znanka**, igralec  $B$  pa poskusi z besedo **zarana**. Igralec  $A$  se odzove takole:

- Najprej ugotovi, da se para črk na indeksih 0 in 5 ujemata, zato nastavi  $z_0$  in  $z_5$  na  $+$ . Obe črki v obeh besedah prečrta, tako da beseda  $a$  postane **\_nank\_**, beseda  $b$  pa **\_aran\_**.
- Sedaj ugotovi, da je črka  $b_1$  (**a**) prisotna v besedi  $a$  (na indeksu 2), zato nastavi  $z_1$  na  $o$ , črki  $b_1$  in  $a_2$  pa prečrta. Beseda  $a$  je sedaj **\_n\_nk\_**, beseda  $b$  pa **\_\_ran\_**.
- Zatim ugotovi, da je črka  $b_4$  (**n**) prisotna v besedi  $a$  (na indeksu 1), zato nastavi  $z_4$  na  $o$ , črki  $b_4$  in  $a_1$  pa prečrta. Beseda  $a$  se tako prelevi v **\_\_\_nk\_**, beseda  $b$  pa v **\_\_ra\_\_**.
- Na preostala indeksa (2 in 3) v zaporedju  $z$  postavi znak  $-$ . Zaporedje  $z$  je torej enako **+o--o+**.

---

<sup>1</sup>Javanski razred, ki implementira vmesnik **Stroj**. Več o tem boste lahko prebrali v nadaljevanju.

## Vaša naloga

Na tekmovanju bo ogrodje nastopalo kot igralec *A*, vaš stroj pa kot igralec *B*. Ogrodje si bo torej izbralo besedo dolžine  $n$  (na tekmovanju bo  $n = 6$ ) iz vnaprej znanega slovarja (to bo datoteka `slovar.txt`), vaš stroj pa bo moral besedo uganiti.

Svoj stroj izdelajte kot razred z imenom `Tekm_vvvvvvvv`, pri čemer `vvvvvvvv` zamenjajte s svojo vpisno številko. Razred mora implementirati vmesnik `Stroj`. Kot že vemo, to pomeni, da mora implementirati obe njegovi metodi:

- `public abstract void inicializiraj(Set<String> besede):`

Sprejme množico besed, ki tvorijo slovar. Ogrodje bo to metodo poklicalo le ob zagonu.

- `public abstract String poteza(List<Character> odziv):`

Ogrodje bo to metodo poklicalo vsakokrat, ko bo vaš stroj na potezi za tvorbo besede. Ob prvem poskusu bo parameter `odziv` enak `null`, v vseh nadaljnjih poskusih pa bo ta parameter seznam  $n$  znakov iz množice  $\{+, o, -\}$ , ki podaja odziv na predhodno strojevo potezo. Ko bo stroj besedo uganil, bo ogrodje še enkrat poklicalo njegovo metodo `poteza`. Parameter `odziv` bo takrat seznam  $n$  znakov `+`.

Metoda mora v vsakem poskusu vrniti besedo dolžine  $n$  (torej 6). Beseda mora biti sestavljena samo iz malih črk slovenske abecede brez šumnikov, ni pa treba, da pripada slovarju. Ko je parameter `odziv` seznam  $n$  znakov `+`, mora metoda vrniti vrednost `null`.

## Slovar

Celoten slovar je podan v datoteki `slovar.txt`. Ta slovar ni namenjen samo preizkušanju vašega stroja, ampak ga bomo uporabili tudi na samem tekmovanju. Slovar je pridobljen iz Spletnega seznama besed slovenskega jezika<sup>2</sup>, vendar pa smo ohranili le besede dolžine 6, ki vsebujejo zgolj male črke slovenske abecede brez šumnikov.

## Ogrodje

Ogrodje je sestavljeno iz datotek `TestirajEno.java` in `TestirajVse.java` ter pomožne datoteke `TestSkupno.java`. Datoteki `TestirajEno.java` in `TestirajVse.java` sta namenjeni testiranju vašega stroja:

- Program `TestirajEno` preizkusi podani stroj s podano besedo iz slovarja. Poženemo ga takole:

```
java TestirajEno imeStroja besedaIzSlovarja
```

Če bi želeli vzorčni stroj `Tekm_12345678` preizkusiti z besedo `znanka`, bi to storili takole:

```
java TestirajEno Tekm_12345678 znanka
```

- Program `TestirajVse` preizkusi podani stroj z *vsemi* besedami iz slovarja. Program premeša besede v vrstnem redu, kot ga (posredno) določa podano seme naključnega generatorja, nato pa podani stroj po vrsti preizkusi z vsako besedo posebej. Program poženemo takole:

```
java TestirajVse imeStroja semeNakljucnegaGeneratorja
```

---

<sup>2</sup><http://bos.zrc-sazu.si/besede.html>

S sledečim ukazom preizkusimo vzorčni stroj `Tekm_12345678`, pri čemer kot seme naključnega generatorja podamo število 12345:

```
java TestirajVse Tekm_12345678 12345
```

## Primer implementacije stroja: razred `Tekm_12345678`

Pri programiranju svojega stroja lahko izhajate iz razreda `Tekm_12345678`. Gre za povsem delujočo implementacijo razreda `Stroj`. Stroj vzdržuje množico vseh besed iz slovarja, ki so združljive z dosedanjimi odzivi, in v vsaki potezi izbere eno od besed iz te množice (prvo po abecedi). Razred ni slab (datoteka `rezultat_12345678_12345.txt` podaja rezultat zagona ukaza `TestirajVse Tekm_12345678 12345`), a prepričani smo, da lahko dosežete še precej boljši rezultat.

## Potek tekmovanja

Vsako implementacijo stroja, ki jo bomo prejeli, bomo preizkusili z določenim številom ( $\geq 10$ ) zagonov programa `TestirajVse`, vsakokrat z drugačnim semenom. Seveda bomo za vse prispele stroje uporabili isto množico semen.

Pri vsakem zagonu programa `TestirajVse` se bo število točk izračunalo kot vsota števila točk za posamezne besede. Število točk za posamezno besedo bo načeloma enako številu poskusov, ki jih bo vaš stroj potreboval, da bo uganil to besedo. No, če stroj ne bo uganil besede niti v 100 poskusih, bo za tisto besedo prejel 200 točk. Isto število točk bo prejel, če bo izbral besedo napačne dolžine ali besedo z neveljavnim znakom ali pa če metoda `poteza` ne bo vrnila vrednosti `null` takrat, ko bi morala.

**Vsak zagon programa `TestirajVse` bo omejen na 10 minut.**<sup>3</sup> Če bo postopek trajal dlje, bomo vašemu stroju pripisali 200 točk za vsako besedo iz slovarja.

Skupno število točk, ki jih bo prejel vaš stroj, bo enako vsoti števila točk za vsak posamezni zagon.

## Omejitve

- Uporabljate lahko samo razrede iz paketov `java.lang`, `java.util`, `java.util.function` in `java.util.stream`.
- Stroj ne sme tvoriti procesov ali niti.
- Stroj ne sme brati ali izdelovati datotek.

## Nagrade

Avtor prvouvrščenega stroja (tistega z *najmanj* točkami) bo pri predmetu Programiranje 1 neposredno prejel oceno 10 in bo s tem oproščen opravljanja izpita (tako pisnega kot ustnega). Avtorji, ki bodo zavzeli mesta od drugega do desetega, pa bodo na izpitu prejeli dodatne točke:

---

<sup>3</sup>Na to omejitev vas še posebej opozarjamo, saj je ne preverjamo v okviru ogrodja. Enostavneje (in bolj smiselno) jo je namreč preverjati v okviru lupinske skripte, ki bo poganjala ogrodje.

Mesto	Dodatne točke
2.	25
3.	20
4.	15
5.	10
6.–10.	5

Na izpitu bo možno zbrati 100 točk, zato se bodo nagrade lahko kar pošteno občutile.

### Oddaja stroja

Stroj oddajte v obliki ene same datoteke z nazivom `Tekm_vvvvvvvv.java`, pri čemer niz `vvvvvvvv` zamenjajte s svojo vpisno številko. (Morebitne dodatne razrede definirajte kot statične notranje razrede v razredu `Tekm_vvvvvvvv`.)