

# Programiranje 1 — tekmovanje

Rok za oddajo: nedelja, 15. januarja 2023

## Skriti program

»Na Učilnici boste pod rubriko *Tekmovanje* našli datoteko `SkritiProgram.class`.«

»Kako, prosim? Pika-class?!«

»Da, prav ste slišali! Pika-class, ne pika-java!«

»Hm ... Ali to pomeni, da se bomo šli detektive? Da bomo morali odkriti skriti program, ki se prevede v datoteko `SkritiProgram.class`?«

»V bistvu ... da in ne! Vaša naloga je, da na Učilnico do nedelje, 15. januarja, oddate *čim manjšo* datoteko `.class`, ki je enakovredna datoteki `SkritiProgram.class`.«

»Ali to pomeni, da bo morala moja datoteka `.class` pri vseh vseh vhodih tvoriti iste izhode kot datoteka `SkritiProgram.class`? Lahko morda poveste kaj več o vhodu, ki ga sprejme ta skriti program, in izhodu, ki ga proizvede?«

»Program na vhodu sprejme dve celi števili z intervala  $[1, 10^6]$ , ločeni s presledkom, izpiše pa eno samo celo število. No, če smo natančnejši, program povsem pravilno deluje tudi za večja vhodna števila, vendar pa vam jamčimo, da bosta v vseh testnih primerih, s katerimi bomo preverjali vašo datoteko `.class`, vhodni števili pripadali intervalu  $[1, 10^6]$ .«

»Datoteke `.class` so vezane na različico prevajalnika in izvajalnika, mar ne?«

»Drži! Datoteka `SkritiProgram.class` je bila izdelana s prevajalnikom različice 17. Če vaš prevajalnik pripada novejši različici, morate svoj program prevesti takole (morda boste dobili kakšno opozorilo, a se ne vznemirjajte):

```
javac -source 17 -target 17 DNTE_vvvvvvvv.java
```

Če številka vaše različice prevajalnika ni večja od 17, pa vam ni treba storiti ničesar. In ja, mimogrede, ime oddane datoteke naj bo `DNTE_vvvvvvvv.class`, pri čemer `vvvvvvvv` nadomestite s svojo vpisno številko. Ponavljamo: pika-class, ne pika-java!«

»Kako določim različico javanskega prevajalnika?«

»Enostavno:«

```
javac -version
```

»Če odprem datoteko `SkritiProgram.class` v besedilnem urejevalniku, dobim same spake. Kako naj se znajdem v tej goščavi?«

»Pomagajte si z orodjem `javap`, ki je standardni del javanske distribucije:

```
javap -c -p SkritiProgram.class
```

Dobili boste kodo v zbirniku JVM. Gre za preprost jezik. Njegov opis najdete v sledečem dokumentu:«

<https://docs.oracle.com/javase/specs/jvms/se17/jvms17.pdf>

»Tega je 624 strani! Ste znoreli?! Za neko brezvezno tekmovanje?! Pojdite se solit!«

»Pomirite se, prosim! Datoteka `SkritiProgram.class` uporablja precej omejen nabor ukazov JVM. Edini objekt, ki ga ustvarimo, je stari dobri bralnik (`Scanner`). Nobene tabele ne izdelamo. Poleg neobhodnega tipa `Scanner` uporabljamo le še tip `int`. Vaša naloga bi bila težko lažja ... «

»No, ja, naj vam bo ... Kakšen pristop pa priporočate?«

»Ena možnost je, da podrobno preučite datoteko `SkritiProgram.class`, napišete javanski program, ki počne enako, potem pa poskušate program preoblikovati tako, da bo ciljna datoteka `.class` čim manjša. Lahko pa se naloge lotite tudi z manipulacijo same datoteke `SkritiProgram.class`. V tem primeru vam morda niti ni treba poznati njenega delovanja; lahko zgolj, recimo, poiščete »suboptimalna« zaporedja ukazov in jih zamenjate z enakovrednimi, a krajšimi zaporedji. Nikakor ne trdimo, da je to mogoče, kaj šele, da je enostavno, vseeno pa naj omenimo, da je eden od vaših kolegov pred kratkim nekaj podobnega naredil s kodo, ki jo tvori sloviti prevajalnik `gcc`. Celo pri največji stopnji optimizacije mu je s transformacijami, ki jih je sam odkril, uspelo dolžino kode zmanjšati za dva odstotka.«

»Ali lahko poleg programa `javap` uporabljam še katero drugo orodje?«

»Ne, žal ne smete. Program `javap` je edino dovoljeno orodje za analizo datotek `.class`. Lahko pa napišete svoj lasten program za delo s temi datotekami. Verjemite mi, veliko se boste naučili.«

»Kako pa boste zagotovili, da ne bodo tekmovalci uporabljali nič drugega kot `javap`?«

»Ne bomo. Zaupamo vam. Če bo kdo do nagrade prišel po nedovoljeni poti, pa bomo držali pesti, da se mu bo zataknila v grlu.«

»Nagrade? Si mar lahko obetamo kakšne nagrade?«

»Seveda, a le v primeru, če bo vaša datoteka `.class` v pravilnem formatu (naj ponovimo: preverjala se bo z izvajalnikom različice 17, ne 19!) in če bo na vseh testnih primerih proizvedla pravilen rezultat (v okviru standardne enosekundne časovne omejitve, seveda). Datoteke, ki bodo ustrezale tema dvema pogojem, bomo razvrstili po naraščajočih velikostih. Avtor prvouvrščene (torej najkrajše pravilne) datoteke bo predmet opravił z oceno 10, ne da bi mu bilo treba opravljati pisni in ustni izpit. Drugouvrščeni tekmovalec bo na pisnem izpitu prejel 25 dodatnih točk, tretjevrščeni 20, četrtouvrščeni 15, petouvrščeni 10, tekmovalci od šestega do desetega mesta pa bodo na pisnem izpitu prejeli po 5 dodatnih točk.«

»Tole se pa lepo sliši! Se že poglobljam v jezik JVM, potem pa se lotim še zgradbe datotek `.class` ... «

»Nas zelo veseli, upamo pa, da vas ne motivirajo le točke ... Veliko užitkov pri programiranju!«