

## Programiranje 2 - pisni izpit

2. 6. 2023 ob 10.00 (čas pisanja: 100 minut)

### Naloga 1 (20 točk)

Kolesarski števec ob cesti šteje, koliko kolesarjev je ta dan peljalo mimo števca. Prikazuje tri podatke: današnje število kolesarjev, skupno število kolesarjev v tem letu in današnji datum. Kolesar Lojze se je nekega lepega večera v ne-prestopnem letu sekundo pred polnočjo peljal mimo kolesarskega števca. Zanimalo ga je, ali je danes mimo števca peljalo več kolesarjev kot druge dni v tem letu. Pomagaj mu napisati ne-statično metodo

```
int kaksenDan(String datum, int danes, int letos)
```

ki prejme datum (v obliki MM.DD - najprej mesec (MM), nato dan (DD)); primer: 04.03 pomeni 3. april) ter današnje (danes) in letošnje (letos) število kolesarjev ter vrne -1, 0 ali 1, če je bil danes podpovprečen, povprečen ali nadpovprečen dan. Dan je »povprečen« (nadpovprečen, podpovprečen), če je tisti dan mimo števca peljalo »enako število« (več, manj) kolesarjev, kot je povprečje:

```
povprečje = število_kolesarjev_do_sedaj / število_dni_do_sedaj.
```

Metodo `kaksenDan()` napiši v razredu `Naloga11`, kateremu dodaj metodo `main()`, ki izpiše rezultat metode `kaksenDan()`, če jo kličemo s prvimi tremi argumenti, podanimi ob klicu programa.

## Naloga 2 (20 točk)

V tekstovni datoteki imamo shranjene podatke o stanju kolesarskega števca za vsak dan neprestopnega leta (imamo torej 365 podatkov). V vsaki vrstici datoteke je zapisan en podatek v obliki

`datum:danesh`

kjer je `datum` oblike `MM.DD`, `danesh` pa število kolesarjev, ki so tisti dan peljali mimo števca. Podatki v datoteki niso zapisani po vrsti (tj. datoteka ni urejena po datumih).

Primer začetka vsebine datoteke (celotna datoteka: `kolo-danesh.txt`)

```
04.30:2018
11.04:2091
03.03:2084
06.24:2039
07.26:2062
```

V razredu `Naloga12` napiši metodo `urediDatoteko(String vhodnaDatoteka)`, ki prebere podatke iz vhodne datoteke in jih v urejenem vrstnem redu (po datumih naraščajoče) zapiše na zaslone. Pri tem naj vsaki vrstici doda še podatek o skupnem številu kolesarjev, ki so do vključno tistega dne peljali mimo števca.

Napiši tudi metodo `main()` in v njej kliči metodo `urediDatoteko()` s prvim argumentom, podanim ob klicu programa.

Primer začetka izhoda (celoten izpis je v datoteki `kolo-letosh.txt`):

```
01.01:1986:1986
01.02:2082:4068
01.03:2056:6124
01.04:1970:8094
01.05:1919:10013
```

...

### Naloga 3 (35 točk)

V tekstovni datoteki imamo shranjene podatke o stanju kolesarskega števca za vsak dan ne-prestopnega leta (imamo torej 365 podatkov). V vsaki vrstici datoteke je zapisan en podatek v obliki

`datum:danes:letos`

kjer je `datum` oblike `MM.DD`, `dan`es število kolesarjev, ki so tisti dan peljali mimo števca in `letos` število kolesarjev, ki so letos (do vključno tega dne) peljali mimo števca. Podatki v datoteki so urejeni po vrsti (po datumih naraščajoče). Primer: `kolo-letos.txt`.

- (10 točk) V datoteki `Naloga13` napiši razred `Stanje`, za shranjevanje podatkov o enem stanju. Razred naj vsebuje privatne attribute `datum`, `dan`es (število kolesarjev danes) in `letos` (število kolesarjev letos do vključno današnjega dne) ter metodo `toString()`, ki izpiše stanje v obliki `»Dan: datum, Danes: danes, Letos: letos«`.
- (10 točk) V razredu `Naloga13` napiši metodo `preberiStanja()`, ki kot parameter prejme ime tekstovne datoteke s stanji. Metoda naj datoteko prebere in vrne seznam (`ArrayList`) vseh prebranih stanj (objektov tipa `Stanje`).
- (12 točk) V razredu `Naloga13` napiši metodo `izpisiTop10()`, ki prejme seznam stanj in na zaslon izpiše 10 najboljših dni (torej 10 dni, ko je mimo števca peljalo največ kolesarjev). Izpis naj bo urejen v naraščajočem vrstnem redu. V primeru enakega števila kolesarjev v enem dnevu naj bo izpis urejen po datumu. Za urejanje podatkov uporabi vgrajeno metodo `sort()`, za podajanje primerjalnika pa uporabi anonimni notranji razred.

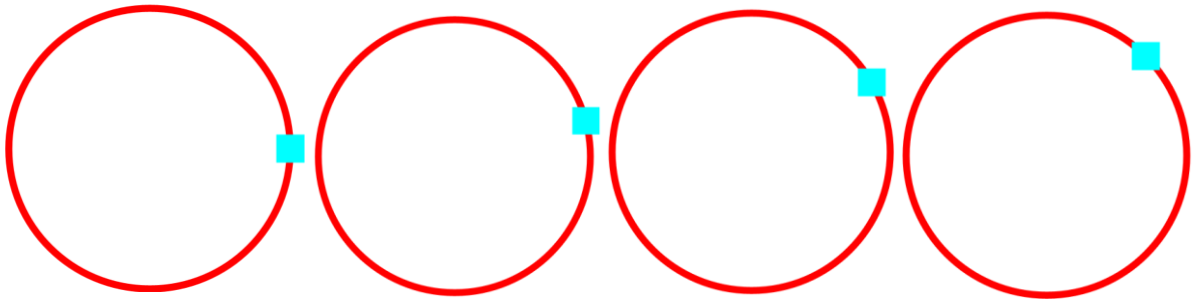
Primer izpisa:

```
Dan: 12.03, Danes: 2093, Letos: 673894
Dan: 02.25, Danes: 2094, Letos: 112171
Dan: 10.15, Danes: 2094, Letos: 576339
Dan: 03.12, Danes: 2096, Letos: 142347
Dan: 08.06, Danes: 2096, Letos: 436182
Dan: 12.25, Danes: 2096, Letos: 717917
Dan: 09.21, Danes: 2097, Letos: 527846
Dan: 03.20, Danes: 2098, Letos: 158382
Dan: 11.26, Danes: 2098, Letos: 660171
Dan: 10.02, Danes: 2100, Letos: 550149
```

- (3 točke) V razredu `Naloga13` napiši metodo `main()`, ki s pomočjo zgornjih metod izpiše najboljših 10 dni iz datoteke, katere ime je podano v prvem argumentu ob klicu programa.

#### Naloga 4 (25 točk)

- a) (15 točk) S pomočjo programa `StdDraw` nariši animacijo, ki prikazuje kroženje modrega kvadratka velikosti  $a \times a$  po krožnici s polmerom  $r = 5 \cdot a$ . Pri vsakem premiku naj se kvadrat premakne za 15 stopinj v obratni smeri urinega kazalca. Kvadrateg naj svoj položaj spremeni 10-krat vsako sekundo. Spodnja slika prikazuje prve štiri položaje kvadratka pri kroženju.



- b) (10 točk) Program dopolni tako, da bo namesto enega krožilo  $m$  kvadratkov, kot je prikazano v filmu (glej `krozenje.mp4`).

Opomba: pri izrisu zgornjih slik smo uporabili koordinatni sistem  $(-100, 100)$ ,  $r = 40$ ,  $\text{kot} = 15$ , debelino svinčnika  $0.01$ , pri filmu pa dodatno  $m = 10$ .