

**Programski jeziki - JAVA**

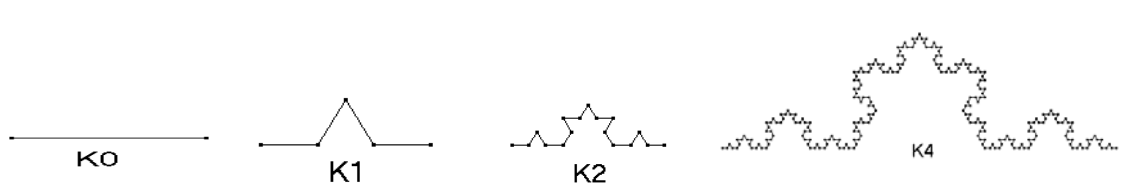
Ime in priimek: \_\_\_\_\_

Točk: \_\_\_\_\_ / 120 (100% = 100 točk)

[20 T] Dan je razred `LogoZelva` z objektnimi metodami `public void fd(int d)`, `public void bk(int d)`, `public void right(int kot)` in `public void left(int kot)`. Prva premakne želvo naprej za `d` enot (pri tem se seveda nariše črta dolžine `d`), druga jo premakne nazaj za `d` enot (tudi tu nastane črta dolžine `d`). Smer želve se pri obeh ukazih ne spremeni. Metoda `right` spremi smer želve za dani parameter stopinj v desno, metoda `left` pa obrne smer želve v levo. Pri teh dveh metodah se položaj želve ne spremeni, pa tudi ne nariše se nič.

Sestavi statično metodo `public static void crtaK(LogoZelvo zelva, int n, int d)` ter nariše sliko, kot je prikazano na spodnjih slikah. Prvi parameter je objekt tipa `LogoZelva` s katerim rišemo, drugi parameter določa red slike (spodnje slike so redov 0, 1, 2 in 4) in tretji dolžino črte (vodoravno). Deli črte `K1` so tretjine vodoravne dolžine.

Namig: preprosta rešitev je rekurzivna (premisli, kako sliko reda  $n$  dobiš iz slik nižjega reda).



[20 T] Janez in Peter sta izumila poseben kodni sistem. Najprej si pošljeta tabelo nizov. Nato si za vsako sporočilo izmenjata seznam števil, izmed katerih vsaka predstavlja besedo, ki je na seznamu nizom številčena s tem številom. Pomagaj jima pri hitrejšem dekodiranju in napiši metodo `public static String dekodiraj(String[] nizi, int[] koda)`, ki sprejme seznam nizov in kodo ter vrne odkodirani niz. Tako del programa

```
String[] nizi
    = {"to", "je", "res", "odlična", "metoda", "za", "kodiranje", "metka", "ni", "slaba", "je", "dobra"};
int mojeSporocilo = {2, 9, 4, 5, 6};
```

```
System.out.println(dekodiraj(nizi, mojeSporocilo);
```

izpiše niz "res slaba metoda za kodiranje".

[20 T] Sestavi metodo `String opisnaOcena(int t, int maxTock)`, ki sprejme število točk `t`, ki jih je dosegel študent in maksimalno število možnih točk `maxTock` in vrne opisno oceno "nezadostno" (0 – 49,999 %), "zadostno" (50 % - 59,999 %), "dobro" (60 % - 69,999 %), "prav dobro" (70 % - 89,999 %) ali "odlično" (90 % ali več).

Na primer `opisnaOcena(63, 100)` vrne niz "dobro".

**[20 T]** V Sloveniji je registrska številka (zaenkrat še) sestavljena iz dveh nizov. Prvi niz je območje in vsebuje dva znaka, drugi niz pa je registracija in vsebuje pet poljubnih znakov. Na primer, registrska številka LJ V1-02E sestoji iz območja LJ in registracije V102E.

(a) Definiraj razred `Registracija`, ki vsebuje podatke o registrski številki avtomobila. Ker se ta načeloma nikoli ne spremeni, naj bo v razredu na voljo le konstruktor, ki sprejme oba niza ter metoda `toString`, ki dano registrsko številko izpiše v obliki `xy_ab-cde`, kjer sta `xy` črki območja, `abcde` pa ustrezni znaki registracije.

(b) V razredu napiši še objektno metodo `boolean veljavnoObmocje()`, ki vrne `true`, če ima registerska številka veljavno območje. V Sloveniji so veljavna območja LJ, KR, KK, MB, MS, KP, GO, CE, SG, NM in PO.

Primer uporabe:

```
Registracija x = new Registracija("MM", "V102E");
boolean b = x.veljavnoObmocje(); // b == false
```

---

**[20 T]** Napiši program `Razmakni.java`, ki iz ukazne vrstice sprejme argument in ga izpiše tako, da vrine med znake besede presledke. Primeri:

```
> java Razmakni prestolonaslednik
p r e s t o l o n a s l e d n i k
> java Razmakni Zakuga?
Z a k u g a ?
> java Razmakni brez muje se se cevelj ne obuje
b r e z
> java Razmakni "brez muje se se cevelj ne obuje"
b r e z m u j e s e s e c e v e l j n e o b u j e
```

Opomba: pred prvim in za zadnjim znakom program ne izpiše presledka.

---

**[20 T]** Karl Gauss pri pouku ni bil priden, zato mu je profesorica dala izračunati vsoto prvih 100 naravnih števil. Ker jo je presenetil s hitrim odgovorom, ga je sklenila bolj zaposliti. Sklenila je, naj izračuna vsoto cifer vseh števil, ki jih bo napisala na tablo. Mladi Karl pa ni želel zapravljati časa z računanjem na pamet, zato te je prosil, da mu napiši program v Javi. Gauss bo števila sproti vpisoval v računalnik. Vnos se zaključi s številko 0.

Če bo torej Gauss vnesel števila 123, 56, 11, 0, mora tvoj program kot odgovor vrniti 19 ( $1 + 2 + 3 + 5 + 6 + 1 + 1$ ).

Namig: Sestavi metodo, ki za dano celo število vrne vsoto njegovih števk. Nato to metodo uporabi za vsako vnešeno število!