



Test JAVA

Ime in priimek:

1. Napiši *metodo* `vsotaStevk`, ki za dano celo število vrne njegovo *vsoto števk* (kot tip `int`). Napiši še *metodo* `povprečnaStevka`, ki za dano celo število vrne njegovo *povprečno števko* (kot tip `double`). Program naj vsebuje obe metodi in testno okolje.
2. Napiši program, ki bo vseboval tri metode (vse vrnejo `int`), ki iz niza oblike "dd.mm.lllll" (npr 15.12.1999) izločijo dan, mesec in leto. Pazi: dan in mesec sta lahko predstavljena le s po eno števko. Za leto privzamemo, da je vedno dano kot štirimestno število.
3. Poišči maksimum in minimum tabele števil na naslednji način. Števila razdeli na polovico in na enak način poišči maksimum in minimum obeh polovic. Iskani minimum/maksimum je manjši/večji od minimuma/maksimuma obeh polovic.
4. Sestavi metodo, ki vrne niz, ki v danem parametru (nizu) zamenja številke z besednim opisom ("1" zamenja z "ena" itd.).
5. Napiši program `Kopija`, ki iz ukazne vrstice sprejme niz `s` in pozitivno celo število `k` ter naredi niz, ki je sestavljen iz `k` kopij niza `s`. Na primer `java Kopija bla 4 naj` v spremenljivko `nov_niz` da niz "blablablbla".
6. Napiši program, ki prebere dolžino v milimetrih in jo pretvori v metre, decimetre, centimetre in milimetre.
7. Sestavi metodo, ki prešteje velike črke v nizu.

[12 T] Za vsakega od naslednjih opisov zapiši podpis metode (kakšen tip vrednosti vrača metoda in kakšne argumente sprejme).

- a) Metoda, ki v podani tabeli realnih števil poišče predzadnje število in ga vrne.

- b) Metoda, ki na grafični objekt `g` nariše krog z radijem `a` in središčem v točki `(x, y)`.

- c) Metoda, ki vrne število različnih znakov v ukazni vrstici.

- d) Metoda, ki na zaslon čim lepše izpiše dvojno tabelo celih števil.

[6 T + 10 T] Dana je metoda `naj`:

```
public static int naj(int a, int b){
    if (a <= b) return b;
    return a;
}
```

a) Kaj izpiše naslednji del programa?

```
int[] tt = {12, 61, 97, 12, 34};
System.out.println("Rezultat metode naj je " +
    naj(naj(tt[1], tt[3]), naj(tt[tt.length - 1], tt[2])));
```

Ta del programa izpiše: _____

b) Uporabi metodo `naj` in sestavi metodo `naj3`, ki vrne večje od treh števil.

[15 T] Kakšna je vrednost spremenljivke `izpis` potem, ko izvedemo naslednje vrstice?

```
int izpis = 1;
for(int zunanji=0; zunanji<100; zunanji=zunanji+5) {
```



```
for(int notranji = 9; notranji >=2; notranji--) {  
    izpis = izpis*2;  
    if(izpis >10)  
        break;  
}  
izpis = izpis + 2;  
if(izpis > 20)  
    break;  
}
```

izpis

[15 T] V razredu `Pra` je metoda `public static boolean tuji(int a, int b)`, ki pove, če sta `a` in `b` tuji si števili. S pomočjo te metode napiši metodo `praštevilo(int p)`, ki pove, če je dano število praštevilo. Namig: praštevilo `p` je tuje vsem številom od 2 do `(int)Math.sqrt(p) + 1`.



[15 T] Popravi spodnji program tako, da bo na zaslon izpisal niz »Alibaba«, če jo kličeš s parametrom »Aa11iibbaabbaa«.

```
public static void alibaba(String s) {  
  
    System.out.print(s.charAt(0));  
  
    alibaba(s.substring(1));  
  
}
```

[25 T] Opiši, kaj naredi spodnji del programa?

```
Random r = new Random();  
int[] table = new int[10];  
for(int i=0; i<10; i++)  
{  
    table[i] = r.nextInt(100);  
}  
  
int a, b, c, d;  
boolean goOut = false;  
while(!goOut) {  
    goOut = true;  
    c = r.nextInt(10);  
    d = r.nextInt(10);  
    if (table[c] < table[d])  
        a = table[c];  
    else  
        a = table[d];  
    if (a == table[c])  
        b = table[d];  
    else  
        b = table[c];  
    table[Math.min(c,d)] = a;  
    table[Math.max(c,d)] = b;  
    for(int i=0; i<9; i++) {  
        if(table[i]>table[i+1])  
            goOut = false;  
    }  
}
```

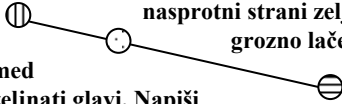
[25 T] Dopolni razred `Zival`, s katerim predstavimo različne živali. Predpostavimo, da moramo za žival poznati njeno najljubšo hrano in število glav. V razredu naj bo en konstruktor, ter metodi s katerima izvemo in nastavimo najljubšo hrano te živali.

```
public class Zival {  
    private String hrana;  
    private int stGlav;  
  
    // konstruktor  
  
    // vrni najljubšo hrano  
  
    // spremeni najljubšo hrano  
  
}
```



[25 T] Iz razreda `Zival` izpelji razreda `TroglaviZmaj` in `Zelenojed`. Troglavi zmaj zna bruhati tudi ogenj do dane razdalje (dano v m) in teči z neko hitrostjo (dano v m/s). Zelenojed ima eno glavo in zna teči z neko hitrostjo (dano v m/s) in zelo rad je zeljnate glave. V obeh razredih potrebujemo metodo s katero izvemo za hitrost živali, v razredu `TroglaviZmaj` pa še metodo s katero izvemo razdaljo, na katero zmaj bruha ogenj.

[25 T] Nekega dne se je `Zelenojed` zerkom sprehajal po prostrani ravnini in pred seboj opazil zeljnato glavo. Žal pa je bil na nasprotni strani zeljnate glave (šrafirane s pikicami na sliki) zlobni troglavi zmaj grozni, ki je bil ravno takrat grozno lačen. Ali zerkom lahko dobi zeljnato glavo, ne da bi ga opekel zmaj grozni? Seveda poznaš razdalje med zeljnati glavi. Napiši zerkom, zeljnato glavo in zmajem groznim. Zmaj in zerkom se premikata direktno proti ustrezen del programskih stavkov.



```
double razdaljaZerkomZeljje = .....;
double razdaljaZmajZeljje = .....;
```

```
if ( _____ ) System.out.println("Zerkom! Navali na zelje!")
else System.out.println("Pazi Zerkom, lahko si pecenka!");
```

[25 T] Napiši program `Zlepek`, ki iz ukazne vrstice sprejme tri nize in jih zlepi (med vsaka dva niza doda po en znak #) ter dobljen niz izpiše na zaslon. Če je uporabnik v ukazno vrstico vnesel premalo parametrov, naj program izpiše prijazno opozorilo. Če jih je vnesel preveč, odvečne zanemari. Na primer:

```
> java Zlepek Zvezdice Zaspanke ni
Zlepljeni niz se glasi: Zvezdice#Zaspanke#ni.
> java Zlepek
Prosim vnesi tri nize.
> java Zlepek Zvezdice Zaspanke ni doma
Zlepljeni niz se glasi: Zvezdice#Zaspanke#ni.
```

[12 T] Na črto napiši, kaj izpiše stavek `System.out.println("Rez = " + h(3) + " / " + h(-3));`

če je metoda `h` taka:

```
public static int h(int n) {
    if (n <= 0) return 1;
    else {
        int res = 5 * h(n-1) - 2;
        return res;
    }
}
```



Podana sta razreda (pri razredu `Izdelek` so eksponenti le oznake za reševanje in NISO del prave kode!)

<pre>public class ZaboJ { protected int x; private int y; public int z; public ZaboJ() { x = 1; y = 2; z = 3; } public ZaboJ(int x, int y, int z) { this.x = x; this.y=y; this.z = z; System.out.println("Ustvarjam zaboJ"); } public ZaboJ obrni() { return new ZaboJ(z,x,y); } public String toString() { return "Mere: " + x + ", " + y + ", " + z; } }</pre>	<pre>class Izdelek extends ZaboJ { private String ime; public Izdelek¹() { ime = ""; } public Izdelek²(String ime) { ime = ime; } public Izdelek³(int x, int y, int z) { ime = ""; this.x = x; this.y = y; this.z = z; } public Izdelek⁴(int x, int y, int z, String ime){ super(x,y,z); this.ime = ime; } public String toString() { return "ZaboJ " + ime + ": " + super.toString(); } }</pre>
------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------	-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------

[10T] Kaj izpiše na zaslon naslednja koda?

```
Izdelek a = new Izdelek("Denarnica");
Izdelek b = new Izdelek(10,20,30,"Miska");
System.out.println(a);
System.out.println(b.obrni().obrni());
System.out.println(b);
```

[5T] Kateri izmed konstruktorjev v razredu `Izdelek` so sintaktično pravilni ?

- 1, 2, 3
- 3, 4
- 1, 4
- 1, 2, 4
- drugo:

[5T] Če v razredu `ZaboJ` spremenimo vse pojavitve besede `protected` v `public`, ali je sedaj sintaktično pravilnih več konstruktorjev ?



Čas reševanja: 90 minut

- DA, 1, 2, 3, 4
- DA, 2, 3, 4
- DA, 1, 3, 4
- DA, 1, 2, 4
- NE
- DA, 1, 2, 3
- DA, 1, 4
- DA, 2, 3, 4.
- drugo: _____

2.) Dana je naslednja metoda:

```
public static void odstevaj(int i) {  
    while (i>0) {  
        for(int j = 1; j<=i; j=j*2) {  
            System.out.print(" " + i + ":" + j);  
            i--;  
        }  
    }  
}
```

[5 T] Kaj izpiše zgornji program, če ga poženemo z argumentom $i = 5$?

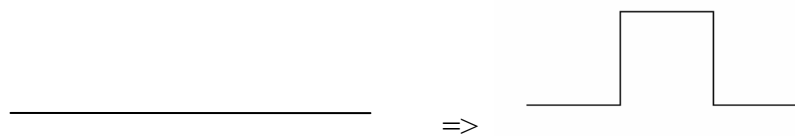
- 5:14:23:12:21:1
- 4:14:23:26:11:1
- 5:14:24:33:22:21:1
- 5:4:3:2:1
- 5:24:45:67:11:1
- drugo: _____.

3.) [20T] Podan je razred LogoZelva, ki pozna metode:

- `public void naprej (int dolzina):` premakne zelvo na dolzina enot naprej.
- `public void nazaj (int dolzina):` premakne zelvo na dolzina enot nazaj.
- `public void levo (int kot):` obrne zelvo levo za kot stopinj.
- `public void desno (int kot):` obrne zelvo desno za kot stopinj.

S pomočjo teh metod napiši metodo `public static void kvadrat (int stopnja, int dolzina, LogoZelva zelva)`, ki nariše kvadrat stopnje $stopnja$ z dolžino stranice $dolzina$. Kvadrat prve stopnje je kar običajen kvadrat. Kvadrat druge stopnje dobimo tako, da kvadratu prve stopnje vsako ravno črto spremenimo po spodnji sliki. Kvadrat stopnje 2 na enak način dobimo iz kvadrata stopnje 1. Namig: napiši metodo, ki pravilno izriše eno stranico kvadrata stopnje n in jih potem pravilno zlepi skupaj.

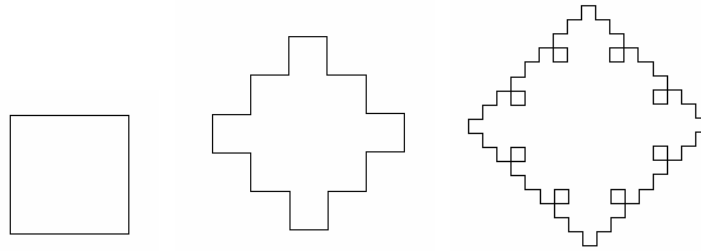
Primer modifikacije ravne črte: na novo dobljeni črti so vse dolžine ravnih črt enake tretjini dolžini začetne črte.



Primeri kvadratov od stopnje 1 do stopnje 3:



Čas reševanja: 90 minut



```
public static void crta (int stopnja, int dolzina, LogoZelva zelva) {  
    ...  
}  
public static void kvadrat (int stopnja, int dolzina, LogoZelva zelva) {  
    ...  
}
```

3.) Dan je naslednji del programa:

```
for (int j = 1; j <= n; j++) {  
    for (int k = 1; k <= n; k = k * 2) {  
        System.out.print(j + " " + k + "/");  
    }  
}
```

[5T] Kaj izpiše gornji del programa, če je n enak 3?

- (a) 1 1/1 2/2 1/2 2/3 1/3 2/
- (b) 1 1/1 2/1 3/2 1/2 2/2 3/3 1/3 2/3 3/
- (c) 1 1/1 2/2 1/2 2/
- (d) 1 1/1 2/1 4/2 1/2 2/2 4/3 1/3 2/3 4/
- (e) nič od zgoraj navedenega

4.) Dana je naslednja metoda, ki ji manjka del kode. Metoda naj bi vrnila tabelo celih števil sum , za katero velja, da je $sum[i] = arr[0] + arr[1] + \dots + arr[i]$. Primer: če je vrednost parametra arr tabela $\{1,4,1,3\}$, naj bi metoda vrnila tabelo $\{1,5,6,9\}$.

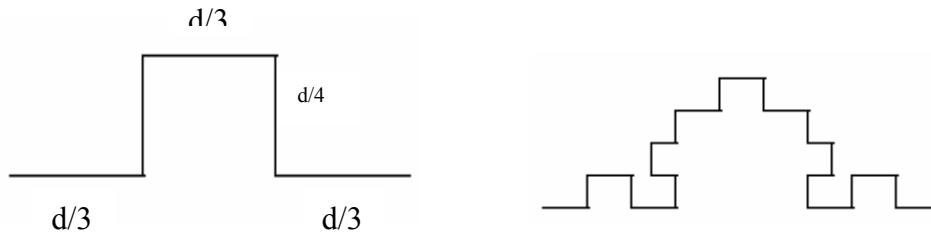
```
public int[] partialSum(int[] arr) {  
    int[] sum = new int[arr.length];  
    for (int j = 0; j < sum.length; j++) sum[j] = 0;  
  
    /* manjkajoca koda */  
  
    return sum;  
}
```

6.) [25T] Dan je razred LogoZelva z metodami $fd(int)$, $bk(int)$, $right(int)$, $left(int)$, ki premaknejo želvo naprej, nazaj, oz. jo zasukajo desno, oz. levo za ustrezeni parameter. S pomočjo objekta iz tega razreda sestavi **metodo** `public static void`



Čas reševanja: 90 minut

$crta(LogoZelva\ ena, int\ n, int\ d)$, ki z želvo ena nariše črto lomljenko stopnje n in dolžine d . Črta stopnje 0 je kar črta v smeri želvinega gibanja dolžine d . Črta stopnje 1 prikazuje prva, črta stopnje 2 pa druga slika.



NALOGA Dana je naslednja metoda

```
public void mystery(int x) {  
    System.out.print(x % 10);  
    if ((x / 10) != 0) {  
        mystery(x / 10);  
    }  
    System.out.print(x % 10);  
}
```

Kaj se izpiše ob klicu $mystery(1234)$?

- (A) 1441
- (B) 3443
- (C) 12344321
- (D) 43211234
- (E) veliko števk zaradi neskončne rekurzije.
- (F) nič od navedenega

NALOGA Dana je naslednja metoda

```
public int checkArray(int[] arr) {  
    int loc = arr.length / 2;  
    for (int k = 0; k < arr.length; k++) {  
        if (arr[k] > arr[loc]) loc = k;  
    }  
    return loc;  
}
```

Kaj najbolje opisuje metodo?

- (A) Vrne indeks prvega elementa v tabeli arr , katerega vrednost je večja kot $arr[loc]$
- (B) Vrne indeks zadnjega elementa v tabeli arr , katerega vrednost je večja kot $arr[loc]$
- (C) Vrne največjo vrednost v tabeli arr
- (D) Vrne indeks največje vrednosti v tabeli arr
- (E) Vrne indeks največje vrednosti v drugi polovici tabele arr



Čas reševanja: 90 minut

NALOGA: Dana je metoda

```
public int mystery(int low, int high, int num, int[] arr) {  
    // pogoj: arr ne vsebuje podvojenih elementov,  
    // podatki v arr so urejeni,  
    // 0 <= low <= arr.length;  
    // low - 1 <= high < arr.length  
  
    int mid = (low + high) / 2;  
    if (low > high) {  
        return low;  
    }  
    else if (arr[mid] < num) {  
        return mystery(mid + 1, high, num);  
    }  
    else if (arr[mid] > num) {  
        return mystery(low, mid - 1, num);  
    }  
    else // arr[mid] == num  
    {  
        return mid;  
    }  
}
```

Kaj vrne klic `mystery(0, arr.length - 1, num, arr)` ?

- (A) Število podatkov v tabeli `arr`, ki so manjši od `num`
- (B) Število podatkov v tabeli `arr`, ki so manjši ali enaki `num`
- (C) Število podatkov v tabeli `arr`, ki so enaki `num`
- (D) Število podatkov v tabeli `arr`, ki so večji kot `num`
- (E) Indeks srednjega elementa v `arr`

Naloga

```
public static boolean metoda(int[] a, int x) {  
    int levo = 0;  
    int desno = a.length - 1;  
    while (levo <= desno) {  
        int sredina = (levo + desno)/2;  
        if (a[sredina] == x) {  
            return true;  
        }  
        else if (a[sredina] > x) {  
            desno = sredina - 1;  
        }  
        else {  
            levo = sredina + 1;  
        }  
    }  
}
```



Čas reševanja: 90 minut

```
return false;  
}
```

Kaj dela metoda? Kakšni morajo biti podatki v tabeli a, da je metoda smiselna?

Naloga 2

V trgovini imajo za vsak izdelek, ki ga prodajajo, naslednje podatke:

- ime
- cena
- ime proizvajalca
- število takih izdelkov na zalogi

Prav tako vsakemu izdelku, ki ga začno prodajati, dodelijo ustrezno zaporedno številko. Definiraj razred `Izdelek`, ki je primeren za predstavitev teh podatkov v Javi. Napiši ustrežni konstruktor. Napiši objektno metodo `toString()`, ki vrne niz, primeren za izpis na ekran, objektno metodo `dobava(int x)`, ki poveča zalogo izdelka za `x` in metodo `povejStevIzdelkov()`, ki vrne število izdelkov, ki jih prodajajo v trgovini. Na primer,

```
Izdelek mleko = new Izdelek("mleko 3.2 1L",  
                             134.50, "Ljubljanske mlekarne", 357);  
mleko.dobava(20);  
System.out.println(mleko.toString());
```

izpiše (predpostavimo, da je mleko prvi izdelek, ki ga začno prodajati) na ekran:

```
Zap.št. izdelka: 1 ::: mleko 3.2 1L (Ljubljanske mlekarne), cena: 134.50 SIT, zaloga: 377
```

Naloga

Dana je metoda `dvojisko(int n)`,

```
public static String kajDelam(int n)  
{  
    String kaj = "";  
  
    while (n > 0)  
    {  
        kaj = (k % 2) + kaj;  
        n = n / 2;  
    }  
  
    return kaj;  
}
```

- Kaj vrne metoda ob klicu `kajDelam(14)`
- Kaj pa ob klicu `kajDelam(8)`
- Kaj dela metoda v splošnem?

Naj bo razred `Ticka` definiran takole:



```
public class Tocka {
    private int x;
    private int y;

    public Tocka()
    {
        this(0, 0);
    }

    public Tocka(int x, int y)
    {
        this.x = x;
        this.y = y;
    }

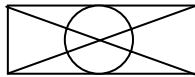
    public int vrni_x()
    {
        return x;
    }

    public int vrni_y()
    {
        return y;
    }

    public void nastavi(int x, int y)
    {
        this.x = x;
        this.y = y;
    }
}
```

Napiši razred `Krog`, ki je definiran s središčem, polmerom in s svojo zaporedno številko, ter naslednje konstruktorje in metode zanj:

- konstruktor `Krog(Tocka sredisce, int polmer);`
- konstruktor `Krog(Tocka levoZgoraj, Tocka desnoSpodaj);` // naredi največji možni krog, ki leži znotraj območja, ki ga določata točki `levoZgoraj` in `desnoSpodaj` in s središčem v preseku diagonal - glej sliko



- metoda `povrsina()`, ki vrne površino kroga (kot dec. število);
 - metoda `prestavi(int gor, int desno)`, ki središče kroga prestavi gor in desno;
 - metoda `povecaj(double faktor)`, ki radij kroga poveča za faktor `faktor`
 - metoda `prestej()`, ki vrne število ustvarjenih krogov
-