



---

Državni izpitni center

---



M 2 3 2 7 8 1 1 3

JESENSKI IZPITNI ROK

# RAČUNALNIŠTVO

---

---

NAVODILA ZA OCENJEVANJE

Torek, 29. avgust 2023

---

SPLOŠNA MATURA

---

Moderirana različica

## IZPITNA POLA 1

Naloga	Točke	Rešitev	Dodatna navodila
1	2	♦ npr. psevdokoda, diagram poteka, zapis v programskem jeziku	Vsak od dveh 1 točka. Upošteevamo tudi druge smiselne odgovore.
2	1	♦ A	
3	1	♦ B	
4	3	<pre> public class test{     public static void main(String[] args){         int x = 0;         int y = -3;         if (Math.abs(y) &gt; 2)             if (y &lt; 0)                 x = 1;             else                 x = 2;         }     } x = 1 y = -3 </pre>	Main in prireditve 1 točka, IF-stavka 1 točka, pravilen izračun obeh spremenljivk 1 točka. Upošteevamo tudi pravilno rešitev brez metode abs.
5	1	♦ A	
6	1	♦ C	
7	1	♦ 4	
8	1	<pre> for(int i=0; i&lt;42; i++)     System.out.println(""+i); </pre>	
9	2	<pre> for (double kot=-2*Math.PI; kot&lt;=2*Math.PI; kot=kot+Math.PI/16)     System.out.println ("kot= "+kot+" sin(kot) = "+Math.sin(kot)); </pre>	Zanka 1 točka, izračun in izpis 1 točka.
10	2	♦ 8 ♦ 8	Vsak pravilen odgovor 1 točka.

Naloga	Točke	Rešitev	Dodatna navodila
11	2	<pre> ♦ double tabela[] = new double[100]; for(int i=0; i&lt;tabela.length; i++)   tabela[i] = Math.random(); </pre>	Ustvarjanje tabele in sprehod po elementih 1 točka, tvorjenje ustreznega naključnega realnega števila 1 točka.
12	2	♦ A, D	Vsak pravičen odgovor 1 točka, že en nepravilen 0 točk.
13	3	♦ -20 ♦ 38 ♦ 4	Vsak pravičen odgovor 1 točka.
14	3	<pre> ♦ public static boolean jeVeljavna(String isbn) {   if (isbn.length() != 13)     return false;   int vs = 0;   for(int i=0; i&lt;13; i++)     if (i % 2 == 0)       vs += (isbn.charAt(i) - '0');     else       vs += 3*(isbn.charAt(i) - '0');   return vs % 10 == 0; } </pre>	Pretvorba števk v vrednost 1 točka, izračun vsote 1 točka, ustrezno vračanje za oba primera 1 točka.
15	1	♦ A	
16	2	<pre> ♦ Scanner vnos = new Scanner(System.in); try{   int stevilo = vnos.nextInt(); }catch(Exception e){   System.out.println("napačen vnos"); } </pre>	Inicializacija Scannerja 1 točka, stavek <i>try-catch</i> 1 točka.
17	2	<pre> ♦ int d = 0; while (a &gt;= b) {   d++;   a -= b; } return d; </pre>	Zanka 1 točka, ustrezen izračun 1 točka.
18	2	<pre> ♦ zdaj.ura= 10 zdaj.min= 35 potem.ura= 12 potem.min= 35 </pre>	Vsaka pravilna vrstica 1 točka.

Naloga	Točke	Rešitev	Dodatna navodila
19.1	4	<pre> public static int[][] spirala(int n) {     int[][] tab = new int[n][n];     int i=0, j=0;     int st=1;     // smeri: 0 - desno, 1 - dol, 2 - levo, 3 - gor     int smer = 0;     while (st&lt;=n*n) {         tab[i][j] = st;         st++;         if (smer==0) {             j++;             if (j&gt;=n    tab[i][j]&gt;0) {                 j--;                 i++;                 smer = 1;             }         } else if (smer==1) {             i++;             if (i&gt;=n    tab[i][j]&gt;0) {                 i--;                 j--;                 smer = 2;             }         } else if (smer==2) {             j--;             if (j&lt;0    tab[i][j]&gt;0) {                 j++;                 i--;                 smer=3;             }         } else if (smer==3) {             i--;             if (i&lt;0    tab[i][j]&gt;0) {                 i++;                 j++;                 smer=0;             }         }     }     return tab; } </pre>	<p>Dodatna navodila</p> <p>Tvorjenje in vračanje tabele 1 točka,  pravilno polnjenje zunanjega roba matrike 1 točka,  pravilno polnjenje v smeri vrstic v notranjosti matrike 1 točka,  pravilno polnjenje v smeri stolpcev v notranjosti matrike 1 točka.</p>

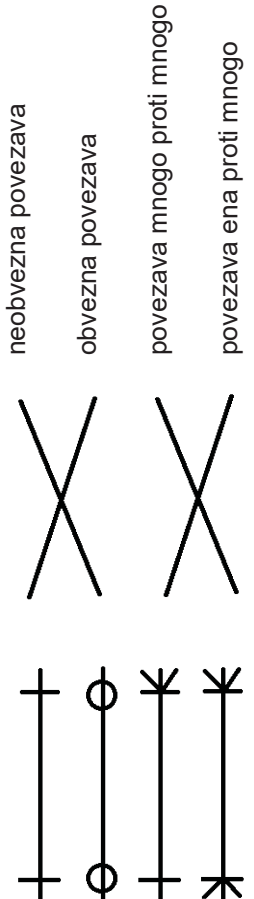
Naloga	Točke	Rešitev	Dodatna navodila
20.1	3	<pre> ♦ static String[] omrezje(String niz[]) {     String tab[] = new String[niz.length];     for (int i=0; i&lt;niz.length; i++) {         niz[i].trim();         StringBuffer sb = new StringBuffer(niz[i]);         boolean brisi = true;         int j = 0;         while (brisi) {             brisi = false;             while (j&lt;sb.length()) {                 if (sb.charAt(j)==' '){                     sb.deleteCharAt(j);                     brisi=true;                 }                 j++;             }             if (sb.charAt(0) == '+')                 tab[i] = "0" + sb.substring(4,6);             else                 if ((sb.charAt(0) == '0') &amp;&amp; (sb.charAt(1) == '0'))                     tab[i] = "0" + sb.substring(5,7);                 else                     tab[i] = sb.substring(0,3);             }         }         return tab;     } </pre>	<p>Brisanje presledkov 1 točka, zanka in vračanje vrednosti 1 točka, pogoj za zajemanje omrežne številke 1 točka.</p>

Naloga	Točke	Rešitev	Dodatna navodila
20.2	1	<pre> ♦ public static void main(String[] args) {     String telefonske[] = {" +38641234567", "00386 31345678",         "040 354666", "070123456"};     String omrezne_st[] = omrezje(telefonske);     for (int i=0; i&lt;omrezne_st.length; i++)         System.out.println(omrezne_st[i]);     } </pre>	

Skupno število točk IP 1: 40

**IZPITNA POLA 2**

Naloga	Točke	Rešitev	Dodatna navodila
1	1	♦ B	
2	1	♦ C	
3	2	♦ D, E	Vsak pravilen odgovor 1 točka, že en nepravilen odgovor 0 točk.
4	2	♦ rm -i besedilo.txt	Ukaz za brisanje 1 točka, stikalo i 1 točka.
5	1	♦ E	
6	1	♦ A	
7	2	♦ plast omrežnega vmesnika (dostopovna plast) ♦ 48 bitov	Vsak pravilen odgovor 1 točka.
8	3	♦ obseg IP-naslovov ♦ poplavni naslov ♦ maksimalno število naprav v prvem podomrežju	En pravilen odgovor 1 točka, dva pravilna odgovora 2 točki, vsi pravilni odgovori 3 točke.
9	2	♦ 9 bitov ♦ 2 zloga	Vsak pravilen odgovor 1 točka.
10	1	♦ D	
11	1	♦ D	
12	2	♦ A, B	Vsak pravilen odgovor 1 točka, že en nepravilen odgovor 0 točk.
13	2	♦ R ♦ N	Vsak pravilen odgovor 1 točka.
14	1	♦ notranji	
15	3	♦ entiteta oz.entitetni tip ♦ razmerje ♦ števnostjo razmerja	Vsak pravilen odgovor 1 točka.

Naloga	Rešitev	Dodatna navodila
<b>16</b> <b>3</b>	<pre> ♦ create table Vprasanje(   zapSt int primary key,   besedilo varchar(100) not null,   tocke int not null);  create table Odgovor(   zapSt int not null,   oznaka char(1) not null,   besedilo varchar(30) not null,   pravilno char(1) not null,   primary key(zapSt,oznaka),   foreign key fk1 (zapSt) references Vprasanje(zapSt)); </pre>	<p>Tabela Vprasanje s PK 1 točka,  tabela Odgovor s PK 1 točka,  FK tabele Odgovor 1 točka.</p>
<b>17</b> <b>2</b>	<pre> ♦ </pre>  <p>neobvezna povezava</p> <p>obvezna povezava</p> <p>povezava mnogo proti mnogo</p> <p>povezava ena proti mnogo</p>	<p>Dve ali tri pravilne povezave 1 točka,  vse pravilne povezave 2 točki.</p>
<b>18</b> <b>2</b>	<pre> ♦ SELECT Potovanje.IDpotovanja, Potovanje.znesek, Stranka.ime FROM Potovanje, Stranka WHERE Potovanje.IDStranke = Stranka.IDStranke AND Potovanje.znesek &gt;= 100 AND Potovanje.znesek &lt;= 500 </pre>	<p>Izbor atributov in povezava tabel 1 točka,  filtriranje 1 točka.</p>

Naloga	Točke	Rešitev	Dodatna navodila
19.1	4		Akterja 1 točka, meje sistema 1 točka, primeri uporabe 1 točka, povezava "extend" 1 točka.

Naloga	Točke	Rešitev	Dodatna navodila
20.1	1	<pre>Update izdelek set cena=cena*0,85 Where zaloga &lt;10;</pre>	

Naloga	Točke	Rešitev	Dodatna navodila
20.2	3	<pre>Select i.izdelekID, sum(i.cena*i.zaloga) From dobavitelj d inner join izdelek i on (d.dobaviteljID=i.dobaviteljID) where d.imeDobavitelja='KupimProdam';</pre>	Povezava tabel 1 točka, pogoj 1 točka, izbor atributov, funkcija in formula 1 točka.

Skupno število točk IP 2: 40