



Državni izpitni center



SPOMLADANSKI IZPITNI ROK

RAČUNALNIŠTVO

NAVODILA ZA OCENJEVANJE

Petek, 7. junij 2013

SPLOŠNA MATURA

Popravljená moderirana različica

IZPITNA POLA 1

Naloga	Točke	Rešitev	Dodatna navodila
1	2	<pre> ♦ public static void main(String[] args) { int y = 0; for (int x=0; x<5; x++) { if (x < y) {y-=3;} else if (x==y) y=2; else {y+=3;} System.out.println(x+" "+y); } } </pre>	<p>Popolnoma pravilna rešitev 2 točki. Zanka 1 točka. Pogoja 1 točka.</p>
2	2	<pre> ♦ for (int i = 1; i < 6; i++) { for (int j = 1; j <= i; j++) { System.out.print(i); } System.out.println(); } </pre>	<p>Zunanja zanka in izpis 1 točka. Notranja zanka in izpis 1 točka.</p>
3	2	♦ A	
4	2	<pre> ♦ a=10 b=25 a=20 b=12 </pre>	<p>Ena vrstica izpisa 1 točka. Obe vrstici izpisa 1 točka. Obe vrstici, v izpisu so navedena le števila 1 točka.</p>
5	2	♦ D	
6	2	♦ C	
7	2	♦ B	
8	2	♦ D	
9	2	♦ B	
10	2	♦ x super.x	<p>Ena vrstica 1 točka. Dve vrstici 2 točki.</p>

Naloga	Točke	Rešitev	Dodatna navodila
11	2	<pre> public class Tabela1 { public static void main(String[] args) { int i, j; int tab[][] = new int[3][5]; for (i=0; i<3; i++){ for (j=0; j<5; j++){ if ((i+j)%2==0) { tab[i][j]=(int) (Math.random() *101); } else { tab[i][j]=100 + (int) (Math.random() *101); } } } for (i=0; i<3; i++){ for (j=0; j<5; j++){ System.out.print(tab[i][j]+" "); } System.out.println(); } } } </pre>	Zanki za polnjenje tabele 1 točka. Pogoj 1 točka.
12	2	<pre> 50 10 10 10 10 </pre>	Prvi element tabele 1 točka, vsi ostali elementi 1 točka.

Naloga	Točke	Rešitev	Dodatna navodila
13.1	4	<pre> public class Nakljucna_vsota120 { public static void main(String[] args) { int a, vsota=0, i=0; while (vsota<120){ a=10 + (int) (Math.random() *11); vsota+=a; System.out.println(a); i++; } System.out.println("Po "+i+" . številu je vsota "+vsota); } } </pre>	Naključna števila iz intervala 1 točka. Zanka in izračun vsote 1 točka. Zanka in štetje števil 1 točka. Program (glava) in izpis 1 točka.

Naloga	Točke	Rešitev	Dodatna navodila
14.1	4	<pre> ♦ public class Zaporedje { public static void main(String[] args) { int n= (int)(Math.random()*21)+10; int[] tabela = new int[n]; int a=10, p=1, clen=0; while (clen<n){ tabela[clen]=a*p; a=a+10; p=p*-1; clen=clen+1; } for (int i=0;i<n;i++){ System.out.print(tabela[i]+" "); } } } </pre>	<p>Naključno število in kreiranje tabele 1 točka. Vnos podatkov v tabelo, če je zaporedje izmenično (+/-), 2 točki, če so elementi zaporedja enako predznačeni, 1 točka. Izpis tabele 1 točka.</p>

Naloga	Točke	Rešitev	Dodatna navodila
15.1	4	<pre> ♦ public static int prestej_besede(String s){ // metoda presteje besede v nizu s (locene s presledki) int i = 0; boolean presledek = true; int st_besed = 0; while(i<s.length()){ if(presledek){ //prejsnji znak je bil presledek if (s.charAt(i) != ' '){ // zacetek nove besede st_besed = st_besed + 1; presledek = false; } } else { // prejsnji znak ni bil presledek if(s.charAt(i) == ' '){ </pre>	<p>Pregled celotnega stavka 1 točka. Preverjanje presledkov in štetje besed z upoštevanjem večkratnih presledkov 2 točki, brez upoštevanja večkratnih presledkov 1 točka. Vračanje vrednosti števila besed v stavku 1 točka.</p>

		<pre> presledek = true; } } i++; } return st_besed; } </pre>	
Naloga	Točke	Rešitev	Dodatna navodila
16.1	2	<pre> ♦ package Atletika; class Tekmovalec { private String ime, priimek, drzava; private double prva; private double druga; private double tretja; Tekmovalec(String i,String pr,String drz,double p,double d,double t) { ime=i; priimek=pr; drzava=drz; prva=p; druga=d; tretja=t; } public String vrniIme() { return this.ime; } public String vrniPriimek() { return this.priimek; } public String vrniDrzavo() { return this.drzava; } public double vrniPrva() { return this.prva; } public double vrniDruga() { return this.druga; } } </pre>	Lastnosti razreda 1 točka. Konstruktor 1 točka.

	<pre> } public double vrniTretja() { return this.tretja; } } </pre>	
--	---	--

Naloga	Točke	Rešitev	Dodatna navodila
16.2	2	<pre> ♦ public class Atletika { static void najBoljsiTekmovalec(Tekmovalec[] tab) { int i=0; double naj=0; double max=0; Tekmovalec t=tab[0]; for(i=0;i<tab.length;i++) { if ((tab[i].vrniPrva()>=tab[i].vrniDruga())&&(tab[i].vrniPrva()>=tab[i].vrniTret ja())){ naj =tab[i].vrniPrva(); } else { if ((tab[i].vrniDruga()>=tab[i].vrniPrva())&&(tab[i].vrniDruga()>=tab[i].vrniTre tja())) { naj =tab[i].vrniDruga(); } else { naj =tab[i].vrniTretja(); } } if (naj>max){ max=naj; t=tab[i]; } } System.out.print("Najboljši tekmovalec je " + t.vrniIme()+" "); System.out.print(t.vrniPrimek()+" "); System.out.println(t.vrniDrzavo()+" "); System.out.println("Najboljši rezultat je "+max); } } </pre>	Iskalni algoritem 1 točka. Dostop do podatkov 1 točka.

Skupno število točk IP 1: 40

IZPITNA POLA 2

Naloga	Točke	Rešitev	Dodatna navodila
1	2	♦ C	
2	2	♦ D	
3	2	♦ B	
4	2	♦ programski števec sklad	
5	2	♦ A	
6	2	♦ C	
7	2	♦ rm /vaje/test/besedilo.doc	Ukaz 1 točka, poštevajo do datoteke 1 točka. Upošteva se tudi druga rešitve za OS MS Windows.
8	2	♦ B	
9	2	♦ D	
10	2	♦ B	
11	2	♦ ip = 01100101.00110110.11111111.00101111 maska = 11111111.11110000.00000000.00000000 omrežje= 01100101.00110000.00000000.00000000 <u>omrežje= 101. 48. 0. 0</u>	
12	2	♦ C	
13	2	♦ A, D	En odgovor 1 točka, oba odgovora 2 točki, že 1 napačen odgovor 0 točk.

Naloga	Točke	Rešitev	Dodatna navodila
14	1	♦ PlaciloID	
	1	♦ Opombe	
Skupaj	2		

Naloga	Točke	Rešitev	Dodatna navodila
15	2	♦ D	

Naloga	Točke	Rešitev	Dodatna navodila
16	1	♦ 7	
	1	♦ 8	
Skupaj	2		

Naloga	Točke	Rešitev	Dodatna navodila
17	2	♦ DISTINCT	

Naloga	Točke	Rešitev	Dodatna navodila
18	1	♦ DA	
	1	♦ DA	
Skupaj	2		

Naloga	Točke	Rešitev	Dodatna navodila
19	2	♦ A	

Naloga	Točke	Rešitev	Dodatna navodila
20	2	♦ A	

Naloga	Točke	Rešitev	Dodatna navodila
21	2	♦ C	

Naloga	Točke	Rešitev	Dodatna navodila
22	2	♦ B	

Naloga	Točke	Rešitev	Dodatna navodila																														
23.1	2	<pre> ♦ Create Table Kraj (KrajID Integer NOT NULL, ImeKraja Char(30) NOT NULL, OpisKraja Varchar(120), Primary Key (KrajID)); </pre>	Stavek CREATE z atributi in tipi 1 točka, integritetne omejitve 1 točka.																														
Naloga	Točke	Rešitev	Dodatna navodila																														
23.2	2	<pre> ♦ SELECT COUNT(*) FROM Hotel h INNER JOIN Kraj k ON (h.KrajID=k.KrajID) WHERE (k.ImeKraja='Ljubljana'); </pre>	Povezovanje tabel 1 točka. Funkcija za štetje 1 točka.																														
Naloga	Točke	Rešitev	Dodatna navodila																														
24.1	4	<table border="1"> <thead> <tr> <th>Pogoji akcije/ukrepi</th> <th colspan="4">Pravila</th> </tr> <tr> <th></th> <th><=3</th> <th>>3</th> <th><=3</th> <th>>3</th> </tr> </thead> <tbody> <tr> <td>Število_dosedanjih_opravljanj</td> <td>D</td> <td>D</td> <td>N</td> <td>N</td> </tr> <tr> <td>Seminarska_naloga_opravljena</td> <td>D</td> <td>D</td> <td>N</td> <td>N</td> </tr> <tr> <td>Prijava_OK</td> <td>N</td> <td>D</td> <td>x</td> <td>x</td> </tr> <tr> <td>Potrebna_komisija</td> <td></td> <td></td> <td></td> <td></td> </tr> </tbody> </table>	Pogoji akcije/ukrepi	Pravila					<=3	>3	<=3	>3	Število_dosedanjih_opravljanj	D	D	N	N	Seminarska_naloga_opravljena	D	D	N	N	Prijava_OK	N	D	x	x	Potrebna_komisija					<p>Pravilno zapisani pogoji 1 točka.</p> <p>Pravilno zapisane akcije 1 točka.</p> <p>Kombinacije pogojev za neuspešno prijavo 1 točka.</p> <p>Kombinacije pogojev za uspešno prijavo 1 točka.</p>
Pogoji akcije/ukrepi	Pravila																																
	<=3	>3	<=3	>3																													
Število_dosedanjih_opravljanj	D	D	N	N																													
Seminarska_naloga_opravljena	D	D	N	N																													
Prijava_OK	N	D	x	x																													
Potrebna_komisija																																	

Skupno število točk IP 2: 52