



Državni izpitni center



M 2 4 2 7 8 1 1 3

JESENSKI IZPITNI ROK

RAČUNALNIŠTVO

NAVODILA ZA OCENJEVANJE

Petek, 30. avgust 2024

SPLOŠNA MATURA

Moderirana različica

IZPITNA POLA 1

Naloga	Točke	Rešitev	Dodatna navodila																												
1	1	♦ B																													
2	1	♦ java -version																													
3	1	♦ D																													
4	2	♦ <table border="1"> <thead> <tr> <th>st</th><th>vs</th><th>x</th><th>x > 0</th></tr> </thead> <tbody> <tr><td>0</td><td>0</td><td></td><td></td></tr> <tr><td>1</td><td>2</td><td>2</td><td>true</td></tr> <tr><td>2</td><td>7</td><td>5</td><td>true</td></tr> <tr><td>3</td><td>13</td><td>6</td><td>true</td></tr> <tr><td>4</td><td>17</td><td>4</td><td>true</td></tr> <tr><td>5</td><td>10</td><td>-7</td><td>false</td></tr> </tbody> </table>	st	vs	x	x > 0	0	0			1	2	2	true	2	7	5	true	3	13	6	true	4	17	4	true	5	10	-7	false	Pravilne vrednosti st in vs 1 točka, pravilne vrednosti x in pogoja 1 točka.
st	vs	x	x > 0																												
0	0																														
1	2	2	true																												
2	7	5	true																												
3	13	6	true																												
4	17	4	true																												
5	10	-7	false																												
5	1	♦ /* moj komentar */																													
6	1	♦ C																													
7	2	♦ continue ♦ break	Vsak odgovor 1 točka.																												
8	1	♦ a = true																													
9	1	♦ Dober dan!																													
10	2	♦ 6, 5 ♦ 6, 5	Vsaka vrstica izpisa 1 točka.																												
11	3	♦ <pre>public static void main(String[] args) { int tabela1[] = {13,18,11,26,33,39,88,52,112,212}; int tabela2[] = new int[tabela1.length]; for(int i=0, j=0; i < tabela1.length; i++) if (tabela1[i] % tabela1[0] == 0) { tabela2[j] = tabela1[i]; j++; } }</pre>	Vsak dopolnjeni izraz 1 točka.																												
12	2	♦ t[0] {3} ♦ t[1] {2,2} ♦ t[2] {1,1,2} ♦ t[3] {0,1,2,3}	Prvi dve vrstici 1 točka, vse vrstice 2 točki.																												

Naloga	Točke	Rešitev	Dodatna navodila
13	2	<pre>♦ for (int i=0; i<niz.length();i++) System.out.println(niz.charAt(i));</pre>	Zanka 1 točka, izpis znaka 1 točka.
14	3	<pre>♦ public static int stPonovitev(String besedilo, char c){ int count=0; for (int i=0;i<besedilo.length();i++){ if (niz.toUpperCase().charAt(i)==Character.toUpperCase(c)) count++; } return count; }</pre>	Glava metode in vračanje vrednosti 1 točka, pregledovanje niza 1 točka, primerjava 1 točka.
15	2	<pre>♦ -3-2-1???Adijo</pre>	Izpis iz zanke 1 točka, izpis iz catch/finally 1 točka.
16	2	<pre>♦ static void izpis() { int t = 0; int sum = 0; int i = 0; while (t < 10*60*1000) { sum += izdelek[i]; t += cas[i]; i++; } System.out.println(sum); }</pre>	Zanka 1 točka, izračun in izpis vsote 1 točka.
17	2	<pre>♦ if (s.length()%2==0) s=s.substring(0,s.length()/2) + "#" +s.substring(s.length()/2,s.length()); else s=s.substring(0,s.length()/2)+"#"+s.substring(s.length()/2+1,s.length() ());</pre>	Pridobivanje podnizov 1 točka, ugotavljanje dolžine niza in združevanje niza 1 točka.

Naloga	Točke	Rešitev	Dodatna navodila
18	3	<pre>♦ private static int indeksNajvecjega(int x[]) { int maxI = x.length-1; for(int i=x.length-1; i>=0; i--) if (x[i] > x[maxI]) maxI = i; return maxI; }</pre>	Iskanje največjega elementa 1 točka, določitev ustreznega indeksa 1 točka in vračanje rezultata 1 točka.

Naloga	Točke	Rešitev	Dodatna navodila
19.1	2	<pre>♦ public static int avtomorfno(int stevilo) { for(int i=stevilo; i>1; i--) if (i%10 == (i*i)%10) return i; return 1; }</pre>	Glava metode in zanka 1 točka, pogoj in vračanje 1 točka.

Naloga	Točke	Rešitev	Dodatna navodila
19.2	2	<pre>♦ public static void avtomorfnisosedii(int tab[][]){ for(int i=0; i<tab.length; i++) for(int j=0; j<tab[i].length; j++) tab[i][j] = avtomorfno(tab[i][j]); }</pre>	Metoda in sprehod po tabeli 1 točka, spreminjanje tabele 1 točka.

Naloga	Točke	Rešitev	Dodatna navodila
20.1	2	<pre>♦ class artikel{ String naziv; double cena; public artikel(String naziv){ this.naziv = naziv; this.cena = 0; } }</pre>	Razred in lastnosti 1 točka, konstruktor 1 točka.

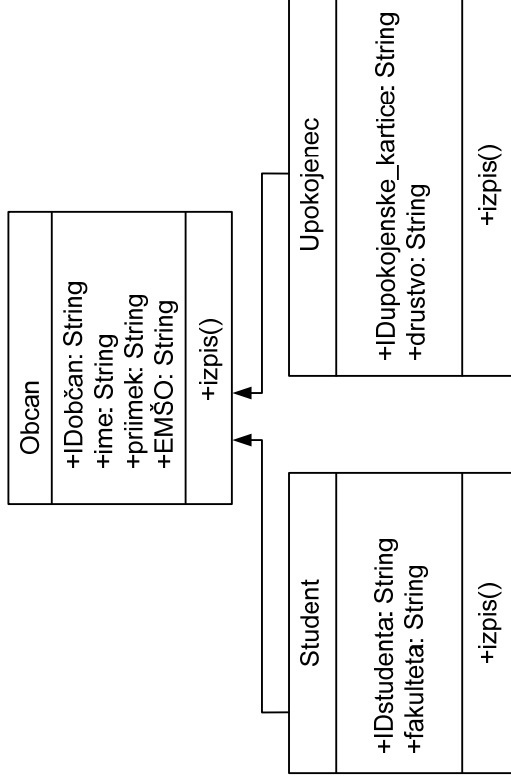
Naloga	Točke	Rešitev	Dodatna navodila
20.2	2	<pre>♦ static void izpis (artikel[] cenik){ double vsota = 0; for (int i=0; i<cenik.length; i++){ vsota +=cenik[i].cena; } double povprecje = vsota/cenik.length; for (int i=0; i<cenik.length; i++){ if (cenik[i].cena >= 1.2*povprecje) System.out.println(cenik[i].naziv + " " + cenik[i].cena); } }</pre>	Glava metode in izračun povprečja 1 točka, pravičen izpis rezultatov 1 točka.

Skupno število točk IP 1: 40

IZPITNA POLA 2

Naloga	Točke	Rešitev	Dodatna navodila									
1	1	♦ C										
2	2	♦ Za odgovor izberejo dve od treh trditev: DRAM: uporablja se za glavni pomnilnik, kapaciteta se izraža v GB, so cenejši od SRAM. SRAM: uporablja se za predpomnilnik mikroprocesorja (<i>cache</i>), kapaciteta se izraža v MB, so dražji od DRAM.	Pravilna poved za DRAM 1 točka, pravilna poved za SRAM 1 točka.									
3	1	♦ -19										
4	2	♦ <table><tr><td>Opis</td><td>Število x</td><td>Število y</td></tr><tr><td>Število x je večje od števila y in ima najbolj desno števko enako F.</td><td>1F</td><td>0F</td></tr><tr><td>Število x in število y sta različni števili in imata zadnji dve števki enaki F.</td><td>Ne obstaja</td><td>Ne obstaja</td></tr></table>	Opis	Število x	Število y	Število x je večje od števila y in ima najbolj desno števko enako F.	1F	0F	Število x in število y sta različni števili in imata zadnji dve števki enaki F.	Ne obstaja	Ne obstaja	Vsak odgovor 1 točka.
Opis	Število x	Število y										
Število x je večje od števila y in ima najbolj desno števko enako F.	1F	0F										
Število x in število y sta različni števili in imata zadnji dve števki enaki F.	Ne obstaja	Ne obstaja										
5	1	♦ mv vaja.txt besedilo.txt										
6	2	♦ Ima dovoljenje za branje, nima ga pa za pisanje in izvajanje. ♦ Dostop se za člane skupine user ne spremeni.	Za prvi odgovor: Če ni zapisano za vse tri, je 0 točk. Vsak pravilen odgovor 1 točka.									
7	1	♦ A										
8	1	♦ predstavitveni sloj										
9	3	♦ Naslov IPv4: 192.168.12.2 ♦ Maska: 255.255.255.0 ♦ Prehod: 192.168.12.1	Vsak pravilni odgovor 1 točka.									
10	1	♦ C										
11	1	♦ B										
12	1	♦ transakcijski informacijski sistem										
13	3	♦ načrtovanje ♦ načrtovanje ♦ implementacija	Vsak pravilni odgovor 1 točka.									

Naloga	Točke	Rešitev	Dodatna navodila
14	2	♦ A, C, E	Dva pravilna odgovora 1 točka, vsi pravilni odgovori 2 točki, samo en pravilen odgovor ali že en napačen 0 točk.
15	2	♦ R ♦ N	Vsak odgovor 1 točka.
16	3	♦ Create table Virus (VID int primary key, imeVirusa varchar(20) not null); Create table Oboleli (Drzava char(3) primary key, VID int not null references Virus (VID), steviloObolelih int not null); ♦ SELECT Dobavitelj.Ime FROM Dobavitelj d, Kategorija k, Artikel a WHERE d.DobaviteljID = a.DobaviteljID AND k.KategorijaID = a.KategorijaID AND k.NazivKategorije = 'kruh' AND d.Posta = 'Ljubljana';	Prva tabela 1 točka, druga tabela 2 točki (1 točka – atributi in ključ; 1 točka – tuji ključ).
17	2		Povezovanje tabel 1 točka, oba pogoja 1 točka.
18	3	♦	Vsak razred 1 točka. Upoštevamo tudi druge smiselne tipe.



Naloga	Točke	Rešitev	Dodatna navodila
19.1	4	<pre>graph LR Vloga_v_filmu -- 1:n --> Film Film -- 1:n --> Igralec Film -- 1:n --> Zvrz Zvrz -- 1:n --> ZvrzFilma</pre>	Entitetni tipi 1 točka, primarni in tuji ključ 1 točka, razmerja in števnosti 1 točka, atributi 1 točka.

Naloga	Točke	Rešitev	Dodatna navodila
20.1	2	<pre>SELECT k.priimek, k.ime, p.imePredmeta, o.tocke FROM kandidat k inner join ocena o on (k.EMSO=o.EMSO) Inner join predmet p on (o.PID=p.PID) Where k.SolaID is NULL Order by k.priimek,k.ime;</pre>	Povezovanje tabel 1 točka, pogoj in razvrščanje 1 točka.

Naloga	Točke	Rešitev	Dodatna navodila
20.2	2	<pre>SELECT s.imeSole, count(k.EMSO) FROM sola s inner join kandidat k on (s.solaID=k.solaID) Group by s.imeSole;</pre>	Povezovanje tabel 1 točka, funkcija in združevanje 1 točka.