



Državni izpitni center



JESENSKI IZPITNI ROK

RAČUNALNIŠTVO

NAVODILA ZA OCENJEVANJE

Četrtek, 27. avgust 2020

SPLOŠNA MATURA

IZPITNA POLA 1

Naloga	Točke	Rešitev	Dodatna navodila
1	2	♦ A, C	Vsak pravilen odgovor 1 točka, že en nepravilen 0 točk.
2	2	♦ double x=(b+Math.sqrt(Math.pow(b,2)-4*a*c))/(2*a)	Uporaba funkcij razreda math 1 točka. Pravilen zapis z oklepaji 1 točka.
3	1	♦ D	
4	1	♦ C	
5	2	♦ a=11 b=10 a=12 b=12	Vsaka spremenljivka 1 točka.
6	2	♦ A:boolean rezultat = (x==y) && (y==z); ♦ B:boolean rezultat = (x%2==0) && (y%2==0) && (z%2==0);	Vsak stavek 1 točka.
7	1	♦ uuu	
8	2	♦ 14 2 28 4 56 8	Vsi a 1 točka, vsi b 1 točka.
9	2	♦ public static void stevila[i]%2 == 0	Vsaka črta 1 točka.
10	1	♦ B	
11	3	♦ import java.util.Scanner; public class Primer{ public static void main(String[] args) { Scanner in = new Scanner(System.in); int X = in.nextInt(); int Vsota=0; while (X>0){ Vsota=Vsota+X%10; X=X/10; } System.out.println(Vsota); } }	Branje in izpis 1 točka. Zanka 1 točka. Izračun vsote posameznih števk 1 točka.

Naloga	Točke	Rešitev	Dodatna navodila
12	3	<pre> ♦ public int indeksNajvecjega() { int idx = 0; for(int i=1; i<prejemki.length; i++) if (prejemki[i]> prejemki[idx]) idx = i; return idx; } </pre>	<p>Glava metode in vračanje 1 točka, sprehod skozi tabelo 1 točka, preverjanje in shranjevanje indeksa največjega elementa 1 točka.</p>
13	3	<pre> ♦ public static void premakniD(int [] x, int n) { if (n<=x.length) { for (int i=x.length-1; i>=n; i--) x[i]=x[i-n]; for (int i=0;i<n;i++) x[i]=0; } premakniD(x,2); } </pre>	<p>Glava metode 1 točka. Premik elementov in vpis 0 na prazna mesta 1 točka. Klic metode 1 točka.</p>
14	1	♦ B	
15	2	<pre> ♦ int x=1; for (int y=0;y<=3;y++) { x=x+y*x; } </pre>	<p>Inicializacija x in y 1 točka, zanka for 1 točka.</p>
16	2	<pre> ♦ public Oseba (String s) { //sestavi osebo iz podatkov, locenih s dvopicji String[] t = s.split(":"); this.ime = t[0]; this.priimek = t[1]; this.dan = Integer.parseInt(t[2]); this.mesec = Integer.parseInt(t[3]); this.leto = Integer.parseInt(t[4]); } </pre>	<p>Funkcija za razdelitev niza 1 točka. Prireditev vrednosti lastnostim 1 točka.</p>
17	1	♦ C	
18	1	♦ D	

Naloga	Točke	Rešitev	Dodatna navodila
19.1	2	<pre> ♦ public class Koda { public static int kljuc; private int vrednost; public Koda() { vrednost = 5678; kljuc=1234; } } </pre>	Razred z lastnostmi (vključno z določili) 1 točka, konstruktor 1 točka.
19.2	2	<pre> ♦ public void kodiraj() { int e = vrednost%10; int d = (vrednost%100)/10; vrednost = (vrednost/100)*100+e*10+d; vrednost += kljuc; } </pre>	Metoda in prištevanje ključa 1 točka. Zamenjava enic in desetic 1 točka.
Naloga	Točke	Rešitev	Dodatna navodila
20.1	4	<pre> ♦ public static boolean preveri(int stevilo) { int zadnja = stevilo%10; int prva = stevilo; while (stevilo!=0) { prva = stevilo; stevilo = stevilo/10; } return prva==zadnja; } boolean z=preveri(34563); </pre>	Glava metode 1 točka. Računanje prve števke 1 točka. Računanje enic in vračanje 1 točka. Klic metode 1 točka.

Skupno število točk IP 1: 40

IZPITNA POLA 2

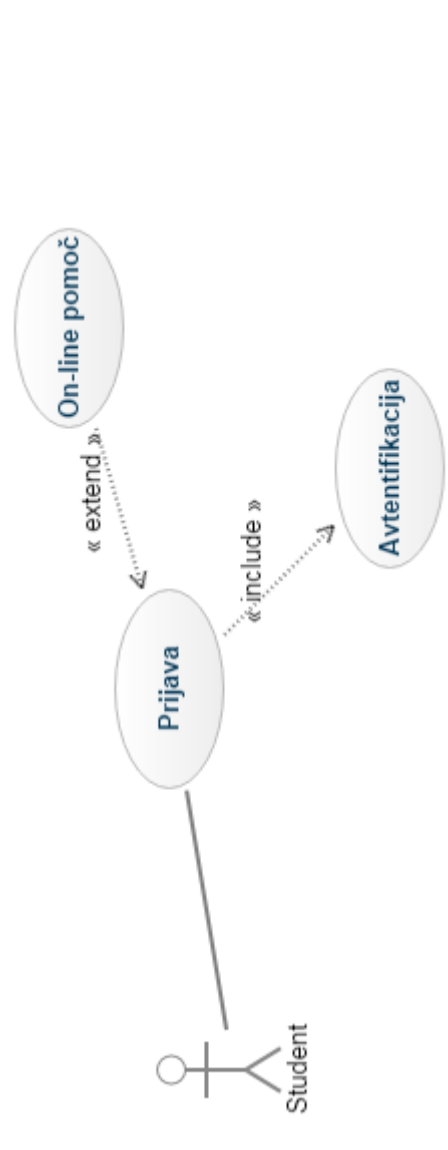
Naloga	Točke	Rešitev	Dodatna navodila
1	2	♦ DEF071 222 240 113	Vsaka vrstica 1 točka.
2	2	♦ N R	Vsaka vrstica 1 točka.
3	1	♦ D	
4	1	♦ C	
5	1	♦ C	
6	2	♦ C, D	Vsak pravilen odgovor 1 točka, že en nepravilen 0 točk.
7	3	♦ 172.16.214.193 255.255.255.192 172.16.214.192	Vsaka vrstica 1 točka.
8	2	♦ B, E	Vsak pravilen odgovor 1 točka, že en nepravilen 0 točk.
9	1	♦ A	
10	2	dve od: ♦ diagram primerov uporabe (use case) ♦ diagram aktivnosti (activity) ♦ diagram zaporedja (sequence) ♦ diagram stanja (statechart) ♦ diagram sodelovanja (collaboration)	Upoštevamo le prva dva zapisana diagrama.
11	1	♦ B	
12	3	♦ EntitetaA EntitetaB KljucB, kljucA	Vsaka vrstica 1 točka.
13	2	štiri od: ♦ Hranijo in obdelujejo veliko količino podatkov. ♦ Obdelava podatkov je enostavna (nezahtevna). ♦ Namenjeni so operativni ravni organizacije. ♦ Odzivni časi so pomembni. ♦ Imajo veliko uporabnikov. ♦ Zahteva se visoka zanesljivost delovanja sistema.	Za dve lastnosti 1 točka. Za štiri lastnosti 2 točki.

Naloga	Točke	Rešitev	Dodatna navodila
14	2	♦ order by Ocena desc	order by 1 točka, desc 1 točka.
15	1	♦ D	
16	2	♦ notranji ali fizični nivo, konceptualni ali srednji nivo in zunanji ali uporabniški nivo arhitektura ANSI/SPARC	Nivoji 1 točka. Ime arhitekture 1 točka.
17	2	♦ B, C	Vsak pravilen odgovor 1 točka, že en nepravilen 0 točk.

Naloga	Točke	Rešitev	Dodatna navodila
18	2	♦ Create table kraj(PostnaSt Char(4) not null, ImeKraja Char(20) not null, primary key(PostnaSt)); Create table oseba(ID Char(8) not null, Ime Char(20) not null, Priimek Char(20) not null, PostnaSt Char(4) not null, Primary key (ID), Foreign key (PostnaSt) references kraj(PostnaSt) on delete no action on update no action);	Vsak stavek create 1 točka. Upoštevam tudi rešitev brez "on delete ...".

Naloga	Točke	Rešitev	Dodatna navodila
19.1	2	♦ Select p.ImePokrajine From Pokrajina p join Drzava d on (p.DID=d.DID) Where d.ImeDrzave='Slovenija' Order by p.ImePokrajine;	Povezava tabel 1 točka. Razvrščanje in pogoj 1 točka.

Naloga	Točke	Rešitev	Dodatna navodila
19.2	2	♦ Select d.ImeDrzave, sum(p.SteviloPrebivalcev) From Drzava d left join Pokrajina on (p.DID=d.DID) Group by d.ImeDrzave;	Povezava tabel 1 točka. Vsota in združevanje 1 točka.

Naloga	Točke	Rešitev	Dodatna navodila
20.1	4	 <pre>graph TD; Student((Student)) --- Prijava((Prijava)); Prijava -.-> «include» Avtentifikacija((Avtentifikacija)); Prijava -.-> «extend» OnLinePomoč((On-line pomoč));</pre>	Akteur 1 točka, use case 1 točka, povezava include 1 točka, povezava extend 1 točka.

Skupno število točk IP 2: 40