



---

**Državni izpitni center**

---



M 0 7 2 7 8 1 1 3

JESENSKI ROK

# **RAČUNALNIŠTVO**

---

---

NAVODILA ZA OCENJEVANJE

**Petek, 31. avgust 2007**

---

**SPLOŠNA MATURA**

---

**IZPITNA POLA 1****SKLOP ARHITEKTURA IN ORGANIZACIJA RAČUNALNIŠKIH SISTEMOV**  
**NALOGE IZBIRNEGA TIPA**

Obkrožite pravilno trditev.

1. Osnovni gradniki računalniškega sistema so (obkrožite najpopolnejši odgovor):

(2 točki)

- A RAM, ROM, CPE;
- B CPE, pomnilnik, disketna enota;
- C monitor, tipkovnica, CPE;
- D CPE, pomnilnik, vhodno-izhodne enote;
- E monitor, ROM, CPE.

2. Pomnilniška hierarhija pomeni določeno organizacijo in medsebojno povezavo pomnilnikov znotraj računalnika. Pomnilniki so v naslednjem vrstnem redu, če gledamo naraščajočo velikost in padajočo hitrost dostopa procesorja:

(2 točki)

- A disk, registri, predpomnilnik, primarni pomnilnik;
- B registri, predpomnilnik, primarni pomnilnik, disk;
- C predpomnilnik, registri, disk, primarni pomnilnik;
- D primarni pomnilnik, registri, predpomnilnik, disk;
- E disk, predpomnilnik, registri, primarni pomnilnik.

3. Zakaj je potreben mehanizem prekinitev s prioritetai?

(2 točki)

- A Zaradi procesov, ki zahtevajo takojšen začetek procesiranja.
- B Zato, da lahko procese razvrstimo skladno z zahtevanim časom začetka procesiranja.
- C Samo mehanizem prekinitev zagotavlja začetek in končanje procesiranja v odvisnosti od prioritete procesov.
- D Samo mehanizem prekinitev zagotavlja končanje procesiranja v odvisnosti od prioritete procesov.
- E Samo ta mehanizem zagotavlja, da dobijo vsi procesi ustrezen čas procesorja.

**4. Dolžina naslovnih registrov procesorja določa:***(2 točki)*

- A največjo točnost (precizijo) računanja;
- B širino podatkovnega vodila med predpomnilnikom in glavnim pomnilnikom;
- C število možnih prekinitev;
- D dolžino besede;
- E obseg naslovnega prostora.

**5. Za registrsko arhitekturo procesorjev RISC je značilno, da:***(2 točki)*

- A program sploh ne uporablja pomnilnika za shranjevanje spremenljivk;
- B prevajalniki lažje prevedejo program, ki ne uporablja pomnilniških naslovov;
- C je arhitektura tako poenostavljena, da procesor v ukazih dela le s kratkimi operandi;
- D pozna le vsebovano naslavljanje;
- E se vse aritmetične in logične operacije izvajajo nad vsebinami registrov.

**6. Preliv (overflow) pri računanju razlikujemo od prenosa (carry), ker:***(2 točki)*

- A se prenos pojavi pri seštevanju, preliv pa pri odštevanju;
- B se prenos pojavi pri računanju s 16-bitnimi števili, preliv pa z 8-bitnimi;
- C se prenos uporablja pri računanju s celimi števili, preliv pa pri računanju z realnimi števili;
- D prenos pomeni, da smo presegli maksimalno vrednost, preliv pa minimalno;
- E preliv upoštevamo le pri računanju s predznačenimi števili.

**7. Preslikava logičnih naslovov v fizične je pri uporabi navideznega pomnilnika potrebna, da:***(2 točki)*

- A se neki del programa lahko naloži na različne naslove v fizičnem pomnilniku;
- B se lahko vsak ukaz ali spremenljivka poljubno premešča po fizičnem pomnilniku;
- C se lahko program (software) poveže s strojno opremo (hardware) na fizičnem nivoju;
- D ni fragmentacije pomnilnika;
- E so deli programa, ki so med seboj povezani, v istem naslovnem področju.

8. Pri katerem izmed danih primerov vsebin je pravilno izračunana prečna in vzdolžna parnost (pariteta), če izberemo liho parnost (pariteto)?

(2 točki)

A ✓

1	1	0	0	1	0	0	1	1
1	0	0	0	1	1	1	1	0
0	0	0	1	1	0	1	0	0
1	1	1	1	1	0	1	1	0
0	1	0	1	1	0	0	0	0

B

1	1	0	0	1	0	0	1	1
1	0	0	0	1	1	1	1	0
0	0	0	1	1	0	1	0	0
1	1	1	1	1	0	1	1	0
1	0	1	0	0	1	1	1	1

C

1	1	0	0	1	0	0	1	0
1	0	0	0	1	1	1	1	1
0	0	0	1	1	0	1	0	1
1	1	1	1	1	0	1	1	1
0	1	0	1	1	0	0	0	1

D

1	1	0	0	1	0	0	1	1
1	0	0	0	1	1	1	1	1
0	0	0	1	1	0	1	0	1
1	1	1	1	1	0	1	1	0
0	0	0	1	0	1	0	1	0

E

1	1	0	0	1	0	0	1	0
1	0	0	0	1	1	1	1	1
0	0	0	1	1	0	1	0	1
1	1	1	1	1	0	1	1	1
1	0	1	0	0	1	1	1	1

**SKLOP PROGRAMSKI JEZIKI IN PROGRAMIRANJE**  
**NALOGE IZBIRNEGA TIPA**

Obkrožite pravilno trditev.

9. Kaj se izpiše med izvajanjem naslednjega zaporedja stavkov:

Pascal:

```
var n,x,i,rezultat : word;
begin
  readln(n);
  readln(x);
  rezultat:=1;
  for i:=1 to n do
    rezultat:=rezultat * x;
  writeln(rezultat);
end.
```

C:

```
#include <stdio.h>
void main()
{
  unsigned int n,x,i,rezultat=1;
  scanf("%d",&n);
  scanf("%d",&x);
  for (i=1; i<=n; i++)
    rezultat*=x;
  printf("%d\n",rezultat);
}
```

(2 točki)

- A Produkt števil rezultat in x.
- B n-ta potenca števila x.
- C Vsota števil n in x.
- D Produkt števil rezultat in i.
- E Vsi večkratniki števila x.

**10. V programskem jeziku Pascal oziroma C sta dani deklaraciji spremenljivk:****Pascal:**

```
var tabela=array[1..10] of integer;
```

**C:**

```
int tabela[10];
```

**Katera trditev je pravilna?***(2 točki)*

- A Za deklaracijo spremenljivke `tabela` je uporabljen enostavni podatkovni tip.
- B Za deklaracijo spremenljivke `tabela` je uporabljen sestavljeni homogeni podatkovni tip.
- C Za deklaracijo spremenljivke `tabela` je uporabljen sestavljeni nehomogeni podatkovni tip.
- D Za deklaracijo spremenljivke `tabela` je uporabljen dinamični podatkovni tip.
- E Navedena deklaracija je neveljavna oziroma nepravilna.

**11. Kaj se izpiše po izvršitvi naslednjega zaporedja stavkov:****Pascal:**

```
var cena:integer;
procedure vnesi_ceno(var nova_cena:integer);
begin
    nova_cena:=nova_cena+20;
end;
begin
    cena:=510;
    vnesi_ceno(cena);
    writeln('Cena je: ',cena,' SIT');
end.
```

**C:**

```
#include <stdio.h>
void vnesi_ceno(int *nova_cena);
int main(){
    int cena=510;
    vnesi_ceno(&cena);
    printf("Cena je: %d SIT \n",cena);
    return 0;
}

void vnesi_ceno(int *nova_cena){
    *nova_cena+=20;
}
```

*(2 točki)*

- A Cena je: 510 SIT
- B Cena je: 520 SIT
- C Cena je: 530 SIT
- D Nič od naštetega.
- E Cena je: 480 SIT

**12. Deklarirane so celoštevilске spremenljivke  $x$ ,  $y$  in  $z$ . Dan je tudi izsek iz programa****Pascal:**

```
y:=37; z:=4;  
x:= y mod z;
```

**C:**

```
y=37; z=4;  
x= y % z;
```

**Kakšna je vrednost spremenljivke  $x$  po izvedbi teh programskih stavkov?***(2 točki)*

- A 374
- B 9
- C 33
- D 4
- E 1

**13. Primerjalnega operatorja  $=$  v programskem jeziku Pascal oziroma  $==$  v programskem jeziku C ne moremo uporabljati za primerjavo vsebin dveh spremenljivk, če sta spremenljivki deklarirani:***(2 točki)*

- A kot predznačeni celi števili;
- B kot nepredznačeni celi števili;
- C kot tabeli;
- D kot znaka;
- E primerjalni operator za ugotavljanje enakosti spremenljivk lahko uporabljamo v vseh zgoraj navedenih primerih.

**14. Deklaracija glave funkcije v programskem jeziku Pascal oziroma prototip funkcije v programskem jeziku C vsebuje (obkrožite vse pravilne odgovore):***(2 točki)**(vsi pravilni odgovori 2 točki, dva pravilna odgovora ena točka, že en nepravilen odgovor nič točk)*

- A ime funkcije;
- B podatkovni tip vrednosti, ki jo funkcija vrne;
- C pomnilniški naslov, na katerem je funkcija;
- D argumente funkcije;
- E blok stavkov, ki opredeljuje funkcionalnost podprograma.

**15. Stavki `case` v programskem jeziku Pascal oziroma stavki `switch` v programskem jeziku C lahko zamenja:**

*(2 točki)*

- A zaporedje prireditev;
- B zaporedje stavkov `for`;
- C zaporedje stavkov `while`;
- D zaporedje stavkov `if`;
- E zaporedje vhodno-izhodnih stavkov.

**16. Izvorna koda programa, napisanega v višjem programskem jeziku, se shrani:**

*(2 točki)*

- A kot binarna datoteka;
- B v formatu, ki je določen s programskim jezikom;
- C v formatu, ki se določi pri shranjevanju datoteke;
- D v neznanem formatu;
- E kot tekstovna datoteka ASCII.



**SKLOP PROGRAMSKI JEZIKI IN PROGRAMIRANJE**  
**NALOGE S KRATKIMI ODGOVORI****17. Kaj izpiše naslednji program?***(2 točki)***Pascal:**

```
Var x,y : Integer;
Begin
  x:= 140; y:=50;
  While (x<>y) Do
  Begin
    If (x>y) Then
      x:=x-y
    Else
      y:=y-x;
    WriteLn(x, '|', y);
  End;
  WriteLn(x);
End.
```

**C:**

```
#include "stdafx.h"
#include <stdio.h>
int _tmain(int argc, _TCHAR* argv[])
{
  int x=140, y=50;
  while (x!=y) {
    if (x>y)
      x=x-y;
    else
      y=y-x;
    printf("%d|%d\n",x,y);
  }
  printf("%d\n",x);
  return 0;
}
```

**Rešitev:**

```
90|50
40|50
40|10
30|10
20|10
10|10
10
```

18. Napišite sled izvajanja funkcije za razvrščanje podatkov preglednice x, če je uporabljen algoritem za razvrščanje z izbiranjem.

(2 točki)

**Preglednica x:**

14	21	26	3	11	12	15
----	----	----	---	----	----	----

**Rešitev:**

<b>3</b>	21	26	14	11	12	15
<b>3</b>	<b>11</b>	26	14	21	12	15
<b>3</b>	<b>11</b>	<b>12</b>	14	21	26	15
<b>3</b>	<b>11</b>	<b>12</b>	<b>14</b>	21	26	15
<b>3</b>	<b>11</b>	<b>12</b>	<b>14</b>	<b>15</b>	26	21
<b>3</b>	<b>11</b>	<b>12</b>	<b>14</b>	<b>15</b>	<b>21</b>	26

19. Napišite program, ki izpiše vsa naravna števila iz intervala [1...1000], ki niso deljiva s 14.

(2 točki)

**Rešitev:**

**Pascal:**

```
Var x : integer;
Begin
  For x:=1 to 1000 do
    If (x mod 14 <> 0 ) then writeln(x);
  End.
```

**C:**

```
#include <stdio.h>

int main()
{
  for (int x=1; x<1001;x++) {
    if (x % 14 != 0) printf("%d ",x);
  }
  return 0;
}
```

**20. Kaj vrne naslednja funkcija, če kot argument  $x$  navedemo:**a)  $x='1234321'$ ;b)  $x='1234567'$ .

(2 točki)

**Pascal:**

```
function pali(x:string):integer;
var i : integer;
    pogoj: boolean;
begin
    i:=1;
    pogoj:=true;
    while (i <= length(x) div 2) and pogoj do
    begin
        if x[i]<>x[length(x)-i+1] then
            pogoj:=false
        else
            inc(i);
    end;
    pali:= integer(pogoj);
end;
```

**C:**

```
#include <string.h>
int pali(char *x)
{
    for (int i=0;i<(strlen(x)/2);)
    { if (x[i]!=x[strlen(x)-i-1])
        break;
      else i++;
    }
    return(i==strlen(x)/2);
}
```

**Rešitev:**a) **funkcija vrne število 1;**b) **funkcija vrne število 0.**

**SKLOP PROGRAMSKI JEZIKI IN PROGRAMIRANJE**  
**STRUKTURIRANE NALOGE**

**21. Dana je deklaracija zapisa:**

**Pascal:**

```
type zapis = record
  oznaka : char;
  kategorija : integer;
  opis : string[10];
end;
```

**C:**

```
struct zapis {
  char oznaka;
  int kategorija;
  char opis[11];
};
```

- a) **Napišite podprogram za vnos podatkov v spremenljivko tipa zapis. Podatke preberite s tipkovnice. Zahtevana glava podprograma oziroma prototip funkcije je:**

**Pascal:**

```
procedure vpisi(var x:zapis);
```

**C:**

```
void vpisi(zapis *x);
```

(2 točki)

- b) **Napišite podprogram, ki spremeni vrednost spremenljivke tipa zapis tako, da ji v podatek kategorija vpiše kodo ASCII podatka oznaka. Spremenljivka naj bo argument podprograma. Zahtevana glava podprograma oziroma prototip funkcije je:**

**Pascal:**

```
procedure spremeni(var x:zapis);
```

**C:**

```
void spremeni(zapis *x);
```

(2 točki)

**Rešitev:**

**Pascal:**

```
procedure vpisi(var x:zapis);
begin
  write('vpisi oznako:'); readln(x.oznaka);
  write('vpisi kategorijo:'); readln(x.kategorija);
  write('vpisi opis:'); readln(x.opis);
end;

procedure spremeni(var x:zapis);
begin
  x.kategorija:=ord(x.oznaka); (* ali x.kategorija:=integer(x.oznaka); *)
end;
```

**C:**

```
#include <stdio.h>
void vpisi(zapis *x)
{
  printf("vpisi oznako:"); scanf("%c",&(*x).oznaka);
  printf("vpisi kategorijo:"); scanf("%d",&(*x).kategorija);
  printf("vpisi opis:"); scanf("%s",(*x).opis);
}
```

```
    printf("vpisi opis:"); scanf("%s", (*x).opis);  
}
```

**ali**

```
#include <iostream.h>  
void vpisi (zapis *x)  
{  
    printf("vpisi oznako:"); cin >> (*x).oznaka;  
    printf("vpisi kategorijo:"); cin >> (*x).kategorija;  
    printf("vpisi opis:"); cin >> (*x).opis;  
}  
void spremeni (zapis *x)  
{  
    (*x).kategorija=(*x).oznaka;  
}
```

22. V datoteki 'placa.dat' so zapisani podatki o plačah uslužbencev. Podatki imajo naslednjo obliko:

EMSO	Priimek	Ime	Placa											
			jan	feb	mar	apr	maj	jun	jul	avg	sept	okt	nov	dec
1401980...	Novak	Miha	155000	160000	210000	180000	175000	215000	200000	...	...	...	...	...
...	...	...	...	...	...	...	...	...	...	...	...	...	...	...
...	...	...	...	...	...	...	...	...	...	...	...	...	...	...
...	...	...	...	...	...	...	...	...	...	...	...	...	...	...

**Deklarirajte zapis oziroma strukturo in tip datotečne spremenljivke.  
Napišite podprogram, ki izpiše znesek največje izplačane mesečne plače.**

*(Deklaracije 1 točka, podprogram 3 točke.)*

**Rešitev:**

**Pascal:**

```

type zapis = record
    emso : string[13];
    priimek: string[20];
    ime: string[10];
    placa:array [1..12] of real;
end;
datoteka = file of zapis;
var f : datoteka;
procedure naj_placa(var f : datoteka);
var x : zapis;
    naj : real;
    i : integer;
begin
    reset(f);
    naj:=0;
    while (not(eof(f))) do begin
        read(f,x);
        for i:=1 to 12 do
            if x.placa[i]>naj then naj:=x.placa[i];
        end;
        writeln('najvisja mesecna izplacana placa=',naj:10:2);
    end;
end;

```

**C:**

```

#include <stdio.h>
struct zapis {
    char emso[14];
    char priimek[21];
    char ime[11];
    float placa[12];
};
void naj_placa (FILE *f)
{ float naj=0;
  int i;
  zapis x;
  fseek(f,0,SEEK_SET);

```

```

fread(&x,sizeof(x),1,f);
while (!feof(f)) {
    for (i=0; i<12; i++) {
        if (x.placa[i]>naj)
            naj=x.placa[i];}
    fread(&x,sizeof(x),1,f);
}
printf("najvisja mesecna izplacana placa= %10.2f",naj);
}

```

23. Na atletskem tekmovanju je v metu krogle nastopilo 12 tekmovalcev. Vsak je imel na voljo šest metov. Rezultati, ki so jih dosegli v posameznih serijah, so shranjeni v tabeli `r`, ki ima 12 vrstic (vsaka vrstica ustreza enemu tekmovalcu) in 6 stolpcev (vsak stolpec ustreza eni seriji metov). Neveljavni meti so zabeleženi kot meti z dolžino 0. Napišite podprogram `povprecjeSerije`, ki za vsako serijo posebej izpiše povprečno dolžino metov. Pri tem naj upošteva samo veljavne mete (tiste z dolžino >0).

Uporabite naslednje deklaracije:

**Pascal:**

```

type tabela=array[1..12,1..6] of real;
procedure povprecjeSerije(r:tabela);

```

**C:**

```

float r[12][6];
void povprecjeSerije(float r[12][6]);

```

(4 točke)

**Rešitev:**

**Pascal:**

```

Type tabela=array[1..12,1..6] of real;
Procedure povprecjeSerije(r:tabela);
var s,st,t : integer;
    p : real;
begin
    for s:=1 to 6 do begin
        p:=0;
        st:=0;
        for t:=1 to 12 do begin
            if (r[t,s]> 0) then begin
                p:=p+r[t,s];
                inc(st);
            end;
        end;
        if (st=0) then
            writeln('serija ',s,' ni imela veljavnih metov')
        else
            writeln('povprecje serije ',s,' je ',(p/st):8:2);
        end;
    end;
end;

```

**C:**

```
#include <stdio.h>
float r[12][6];
void povprecjeSerije(float r[12][6])
{ int s,st,t;
  float p;
  for (s=0;s<6;s++){
    p=0;
    st=0;
    for(t=0;t<12;t++) {
      if (r[t][s]> 0) {
        p=p+r[t][s];
        st++;
      }
    }
    if (st==0)
      printf("serija %d ni imela veljavnih metov \n",s+1);
    else
      printf("povprecje serije %d je %8.2f\n",s+1,p/st);
  }
}
```



24. Dan je niz  $n$ , ki je popačen tako, da je prva črka vsake besede prestavljena na konec besede. Napišite podprogram, ki niz  $n$  pretvori spet v prvotno obliko.

**Primer niza**  $n = \text{"avaJ ej bjektnoo smerjenu rogramskip ezikj"}$  pretvorimo v  $n = \text{"Java je objektno usmerjen programski jezik"}$ .

(4 točke)

**Rešitev:**

**Pascal:**

```
procedure popravi(var n : string);
var n2 : string;
    p, i : integer;
begin
    i:=1;
    n2:=n;
    while (i <=length(n)) do begin
        while ((i<=length(n)) and (n[i]=' ')) do begin
            n2[i]:=n[i];
            inc(i);
        end;
        p:=i;

        while ((n[i]<>' ') and (i<=length(n))) do begin
            n2[i+1]:=n[i];
            inc(i);
        end;
        n2[p]:=n[i-1];
        n2[i]:=n[i];
    end;
    n:=n2;
end;
```

**C:**

```
#include <stdio.h>
#include <string.h>
void popravi(char *n)
{ int p,i=0;
  char *n2;
  strcpy(n2,n);
  while (n[i]!='\0') {
    while ((n[i]==' ') && (n[i]!='\0')) {
      n2[i]=n[i];
      i++;
    }
    p=i;
    while ((n[i]!=' ') && (n[i]!='\0')) {
      n2[i+1]=n[i];
      i++;
    }
    n2[p]=n[i-1];
    n2[i]=n[i];
  }
  strcpy(n,n2);
}
```

## IZPITNA POLA 2

Obkrožite pravilno trditev.

### 1. Pri uporabi protokola CSMA/CD:

(2 točki)

- A prihaja do trkov paketov, kar pa ne zmanjšuje učinkovitosti prenosa, saj kolizijski protokol trke predvideva;
- B ni zmanjšanja učinkovitosti prenosa, če oddaja več oddajnikov hkrati;
- C se zmanjša učinkovitost prenosa pri povečanju prometa;
- D ni mogoča uporaba brezžičnega prenosa;
- E se prenaša paket od postaje do postaje, pri čemer vsaka postaja tudi ojača signal.

### 2. Usmerjevalnik pri usmerjanju datagramov:

(2 točki)

- A uporabi za usmerjanje le glavo datagrama;
- B zaradi varnosti pregleda tudi podatkovno polje in preveri, ali so podatki v njem varni za prenos;
- C obvesti oddajnik, da je sprejel datagram, in ga pošlje v ustrezni smeri;
- D zaradi boljšega izkoristka prenosne linije združi več datagramov, ki imajo isto smer, v večji paket;
- E uporabi glavo, pa tudi rep datagrama, da preveri pravilnost prenosa.

### 3. Pri prenosu elektronske pošte se standardni aplikacijski protokol SMTP uporabi za prenos:

(2 točki)

- A od pošiljatelja do prejemnika pošte;
- B od strežnika pošiljatelja do strežnika prejemnika;
- C od strežnika prejemnika pošte do prejemnika pošte;
- D od pošiljatelja do strežnika prejemnika pošte;
- E paketov na transportni plasti.

**4. Maska ob naslovu IP določa:***(2 točki)*

- A kateri deli naslova IP so v glavi paketa;
- B možne naslove IP;
- C da se lahko za posamezen računalnik uporabita tudi naslova 0.0.0.0 in 255.255.255.255;
- D kateri deli naslova IP se ne uporabljajo pri usmerjanju paketa;
- E največje število, ki se uporabi za naslov IP.

**5. Kateri atribut bi lahko uporabili kot enolični identifikator državljanke/državljana Slovenije:***(2 točki)*

- A ime in priimek;
- B rojstni datum;
- C številko potnega lista;
- D naslov;
- E davčno številko.

**6. Pravica kreiranja in spreminjanja shem baze podatkov (opisov podatkov) je namenjena:***(2 točki)*

- A končnim uporabnikom;
- B programerjem;
- C upravitelju baze podatkov;
- D upravitelju baze podatkov in vodjem projektov;
- E upravitelju baze podatkov in programerjem.

**7. Informacija je:***(2 točki)*

- A množica vsaj treh ali več podatkov;
- B formalna predstavitev podatkov;
- C dekodirani podatek;
- D pomen, ki ga uporabnik pripiše podatkom;
- E vse zgoraj navedeno.

**8. V prvi fazi razvoja novega informacijskega sistema:***(2 točki)*

- A ovrednotimo načrt bodočega sistema;
- B izdelamo načrt bodočega sistema;
- C izberemo potrebno razvojno programsko opremo;
- D naredimo prototip rešitve;
- E zberemo zahteve uporabnikov.

**9. Kateri od navedenih vrst informacijskega sistema zahteva najkrajše odzivne čase?***(2 točki)*

- A Transakcijski informacijski sistem.
- B Menedžerski informacijski sistem.
- C Ekspertni sistemi.
- D Vodstveni sistemi.
- E Sistemi za podporo odločanju.

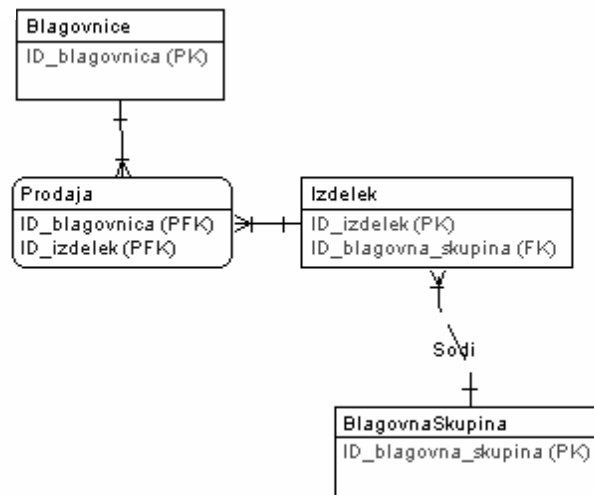
**10. Nepredznačeno dvojiško število 111001,0100 je v šestnajstiškem sestavu:***(2 točki)*

- A 79,4
- B F2,1
- C F9,1
- D 79,2
- E F9,4

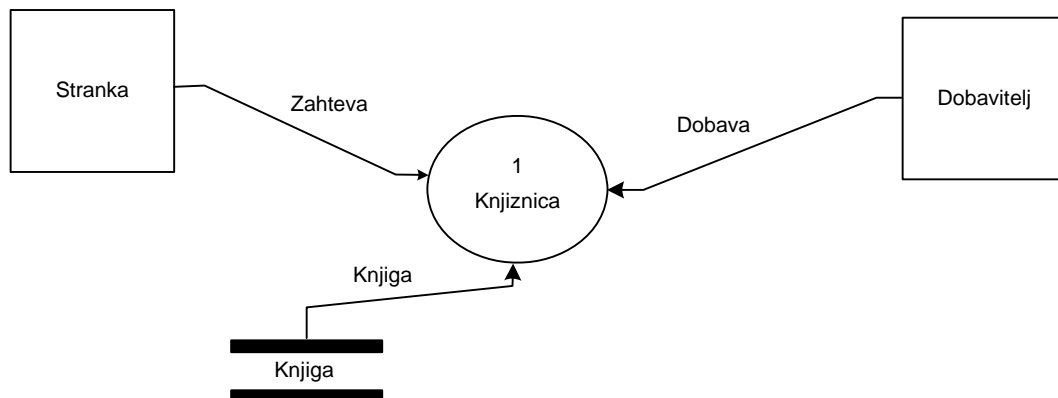
11. Trgovsko podjetje ima več blagovnic, v vsaki pa je naprodaj več izdelkov. Vsak izdelek spada v točno določeno blagovno skupino in je naprodaj v vseh blagovnicah. Narišite entitetno-relacijski diagram, iz katerega bodo razvidna razmerja in števnost med entitetami Blagovnica, Izdelek in BlagovnaSkupina.

(2 točki)

Rešitev:



12. Dan je kontekstni nivo diagrama toka podatkov. Poiščite in pojasnite napaki na diagramu.



(En odgovor 1 točka, oba odgovora 2 točki.)

Rešitev:

proces 'Knjižnica' nima izhodnih tokov, zbirka podatkov se ne sme pojaviti na kontekstnem nivoju DTP.

13. Navedite vsaj dva načina, s katerima se preprečuje nedovoljeni dostop do podatkov.

(2 točki)

**Rešitev:**

gesla, enkripcija podatkov, biometrični podatki, fizične ovire.

14. Kateri stavki SQL sodijo v skupino stavkov SQL DML (Data Manipulation Language)?

(Vsaj dva navedena 1 točka,  
vsi navedeni 2 točki.)

**Rešitev:**

SELECT, INSERT, DELETE, UPDATE.

15. Dane so relacijske sheme in ustrezne tabele:

Proizvajalec (Šifra\_proizvajalca:N, Ime\_proizvajalca:A,  
Naslov\_proizvajalca:A, Kraj\_proizvajalca:A)

Proizvaja (Šifra\_proizvajalca:N, Šifra\_izdelka:N)

Izdelek (Šifra\_izdelka:N, Ime\_izdelka:A, Cena\_izdelka:N, Opis\_izdelka:A)

a) Napišite stavek SQL, ki vrne, koliko izdelkov je zapisanih v tabeli Izdelek.

(1 točka)

**Rešitev:**

SELECT COUNT(sifra\_izdelka)  
FROM izdelek;

b) Napišite stavek SQL, ki izpiše abecedni seznam imen proizvajalcev čokolade.

(3 točke)

**Rešitev:**

SELECT p.ime\_proizvajalca  
FROM proizvajalec p, proizvaja pr, izdelek i  
WHERE (p.sifra\_proizvajalca=pr.sifra\_proizvajalca) and  
(pr.sifra\_izdelka=i.sifra\_izdelka) and  
(i.ime\_izdelka='čokolada')  
ORDER BY p.ime\_proizvajalca;

16. Izdelati želimo informacijski sistem za podjetje XYZ, ki izvaja tečaje tujih jezikov. Sistem bodo uporabljali naslednji uporabniki: organizator tečajev, tečajnik in predavatelj.

Organizator tečajev je oseba, ki je zaposlena v podjetju XYZ in skrbi za vzdrževanje podatkov o posameznih tečajih. Za vsak tečaj vnese šifro tečaja, naziv tečaja, datum začetka in podatke o predavatelju.

Tečajnik je oseba, ki se po internetu prijavi na enega ali več tečajev.

Predavatelj je sodelavec podjetja, ki izvaja enega ali več tečajev. Informacijski sistem mora vsakemu predavatelju omogočati, da si za vsak tečaj izpiše seznam prijavljenih tečajnikov.

Vsak mesec se samodejno sproži postopek za izdelavo računov, ki se izpišejo in po pošti pošljejo posameznim tečajnikom.

- a) Narišite kontekstni nivo DTP.

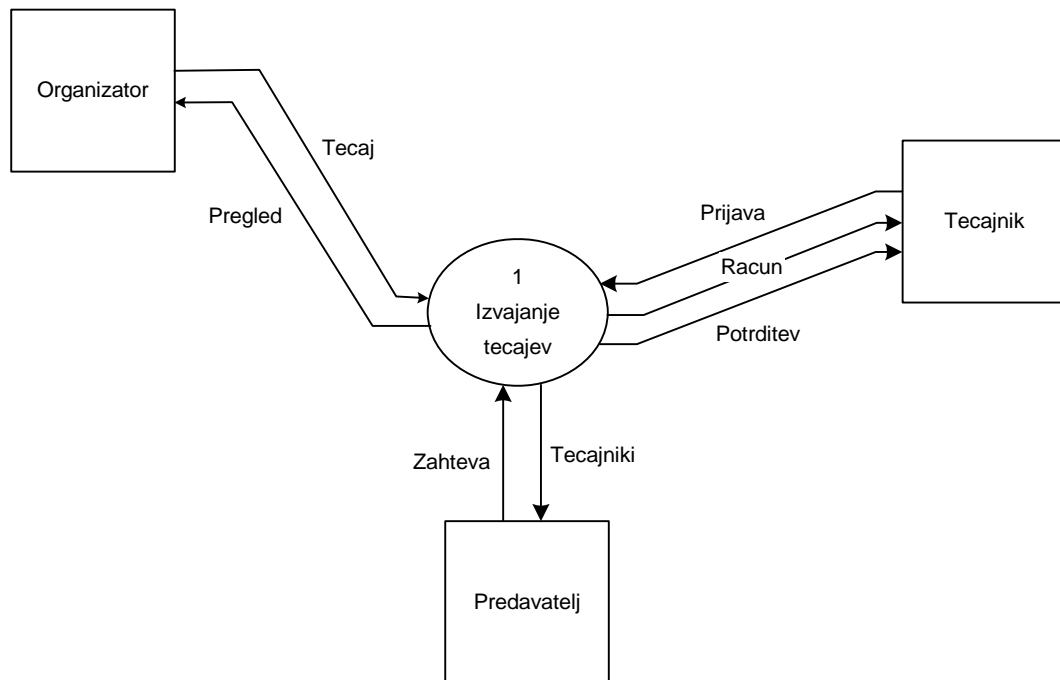
*(1 točka)*

- b) Narišite prvi nivo DTP.

*(3 točke)*

Rešitev:

a)



b)

