



Državni izpitni center



M 0 6 2 7 8 1 1 3

JESENSKI ROK

RAČUNALNIŠTVO

NAVODILA ZA OCENJEVANJE

Torek, 5. september 2006

SPLOŠNA MATURA

Popravljená moderirana različica

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

Obkrožite pravilno trditev.

1. Procesorji z več registri so bolj učinkoviti, ker

(2 točki)

- A program lahko uporablja registre za začasno shranjevanje operandov;
- B se lahko istočasno obdeluje več podatkov;
- C lahko računajo z večjimi podatki;
- D procesor lahko uporablja več različnih načinov naslavljanja;
- E procesor bolje izkoristi aritmetično logično enoto.

2. Če procesor deluje s frekvenco ure 100 MHz, to pomeni, da:

(2 točki)

- A v eni sekundi izvede $100 * 10^6$ ukazov;
- B je en strojni cikel dolg 10 ns;
- C se med CPU in pomnilnikom prenese $100 * 10^6$ informacij;
- D se vsak ukaz izvaja 100 μ s;
- E procesor v eni sekundi obdela $100 * 10^6$ podatkov.

3. Pojem preliv (overflow) se pri računanju povezuje z enim od naslednjih pojmov:

(2 točki)

- A logične operacije;
- B znak (char);
- C šestnajstiški zapis (hexadecimal notation);
- D predznačena števila (signed numbers);
- E vodilo (bus).

4. Za ukaz pogojne vejitve v programu velja:*(2 točki)*

- A omogoča vejitev na N naslovov, odvisno od pogoja;
- B omogoča vejitev na naslov ob izpolnjenem pogoju;
- C nikoli ne pride do vejitve;
- D vejitev se izvede, če je pogojni register v procesorju poln;
- E vejitev se izvede, če naslednji ukaz ni vejitveni ukaz.

5. Za odkrivanje logičnih napak v programu uporabljamo razhroščevalnik (debugger). Njegova vloga je:*(2 točki)*

- A da odpravi napake, ki smo jih zagrešili pri pisanju programa;
- B da simulira delovanje procesorja in pomnilnika;
- C da se program izvede do prve logične napake, ki jo razhroščevalnik samodejno odkrije, nato pa se izvajanje ustavi;
- D da omogoča spremljanje izvajanja programa, prekinjanje izvajanja, vpoglede in posege v pomnilnik in registre;
- E da omogoča prenos programa v pomnilnik, kjer ga lahko zaženemo.

6. Pri takojšnjem naslavljanju je operand:*(2 točki)*

- A spremenljivka;
- B naslov;
- C odmik od naslova;
- D izraz;
- E konstanta.

7. V računalniku se na videz izvaja več procesov hkrati. Da to lahko zagotovimo, moramo imeti:*(2 točki)*

- A več procesorjev;
- B pomnilnik z več vrati;
- C primeren operacijski sistem;
- D procesor z več aritmetično-logičnimi enotami (ALE);
- E dovolj velik sekundarni pomnilnik.

8. V operacijskem sistemu Linux ste prijavljeni kot Uporabnik1. Kakšne morajo biti dovolilnice datoteke prog, da bi jo lahko izvajali vsi uporabniki sistema?

(2 točki)

- A -rwx r-x --- 1 moj group1 20471 feb 4 2000 prog
- B -rwx r-x --x 1 moj group1 20471 feb 4 2000 prog
- C ---x --x --- 1 moj group1 20471 feb 4 2000 prog
- D ---x --x rw- 1 moj group1 20471 feb 4 2000 prog
- E d--x --x rw- 1 moj group1 20471 feb 4 2000 prog

SKLOP PROGRAMSKI JEZIKI IN PROGRAMIRANJE
NALOGE IZBIRNEGA TIPA

Obkrožite pravilno trditev.

9. Na začetku programa so dane naslednje deklaracije:

Pascal:

```
Const cc=99;  
Var x : integer;  
    y : real;
```

C:

```
const cc = 99;  
int x;  
float y;
```

Pri katerem stavku bo prišlo do napake med prevajanjem programa?

(2 točki)

Pascal:

- A x:=x + cc;
- B y:=y + cc;
- C cc:=cc + x;
- D inc(x);
- E dec(x,cc);

C:

- A x=x + cc;
- B y=y + cc;
- C cc=cc + x;
- D x++;
- E --x;

10. Kaj izpiše naslednji izsek programa?*(2 točki)***Pascal:**

```
var x : integer;
begin
x:=3;
case x of
  1 : writeln('1234');
  2 : writeln('234');
  3 : writeln('34');
  4 : writeln('3');
  else writeln('4');
end;
```

C:

```
int x = 3;
switch (x) {
  case 1 : printf("1234");break;
  case 2 : printf("234");break;
  case 3 : printf("34");break;
  case 4 : printf("4");break;
}
```

- A 1234
- B 234
- C 34
- D 3
- E 4

11. Pri kateri od navedenih zank se stavki znotraj zanke izvedejo vsaj enkrat?*(2 točki)***Pascal:**

- A pri zanki repeat.
- B pri zanki while,
- C pri zanki for,
- D pri nobeni zanki,
- E pri vseh treh zankah.

C:

- A pri zanki do while.
- B pri zanki while,
- C pri zanki for,
- D pri nobeni zanki,
- E pri vseh treh zankah.

12. Obkrožite pravilne odgovore.**Za predstavitev algoritma uporabljamo:**

*(2 pravilna odgovora – 2 točki
1 pravilen odgovor – 1 točka
že en nepravilen odgovor – 0 točk)*

- A poslovni model,
- B diagram toka podatkov (DTP),
- C diagram poteka (flow chart).
- D entitetno-relacijski diagram (ERD),
- E psevdokodo.

13. Deklarirani sta tabelarični spremenljivki x in y :

(2 točki)

Pascal:

```
var x,y : array[1..3] of integer;
```

C:

```
int x[3],y[3];
```

S katerim stavkom prepisemo celotno vsebino tabele x v tabelo y ?

Pascal:

A $y:=x$;

B for i:=1 to 3 do y[i]:=x[i];

C $y[1..3]:=x[1..3]$;

D $y[3]:=x[3]$;

E $y[1 \text{ to } 3]:=x[1 \text{ to } 3]$;

C:

A $y=x$;

B for (int i=0;i<=2;i++) y[i]=x[i];

C $y[0..2]=x[0..2]$;

D $*y=x$;

E $*y=*x$;

14. Kakšen je niz x po izvedbi naslednjega programa?

(2 točki)

Pascal:

```
var x:string; y:integer;
begin
  x:='naloga je zelo lahka';
  delete(x,8,2);
  insert('ni',x,8);
  y:= pos('lahka',x);
  delete(x,y,5);
  insert('tezka',x,y)
  writeln(x);
end.
```

C:

```
#include <stdio.h>
#include <string.h>
int main()
{
  char x[256]="naloga je zelo lahka";
  char *p=strstr(x,"je");
  int k=p-x;
  x[k]='\n'; x[k+1]='i';
  p=strstr(x,"lahka");
  k=p-x;
  x[k]='\0';
  strcat(x,"tezka");
  printf("%s",x);
}
```

- A naloga ni težka;
- B naloga je zelo lahka;
- C naloga ni zelo težka;
- D naloga je zelo težka;
- E naloga je težka.

15. Dana je glava podprograma:

(2 točki)

Pascal:

```
procedure abc(var a:integer;b:integer);
```

C:

```
void abc(int *a,int b)
```

Kateri klic podprograma je pravilen?

Pascal:

- A abc(1,1);
- B abc(a,1);
- C write(abc(1,1));
- D abc(1,a);
- E nobeden od klicev ni pravilen.

C:

- A abc(1,1);
- B abc(&a,1);
- C fprintf("%d",abc(1,1));
- D abc(1,a);
- E nobeden od klicev ni pravilen.

16. Dani sta deklaraciji zapisa in datoteke:*(2 točki)***Pascal:**

```
Type x=record
    stev:integer;
end;
dat = file of x;

var
    podatek:x;
    f:dat;
```

C:

```
struct x {
    int stev;
} ;
x podatek;
FILE *f;
```

Kateri od navedenih ukazov za pisanje zapisa na datoteko je pravilen?**Pascal:**

- A write(f,x);
- B writeln(f,x.stev);
- C write(f,podatek);
- D writeln(f,podatek);
- E nobeden izmed predhodno navedenih.

C:

- A fprintf(f,"%d",x);
- B fprintf(f,"%d",x.stev);
- C fprintf(f,"%d",podatek.stev);
- D fprintf(podatek,f,"\n");
- E nobeden izmed predhodno navedenih.

SKLOP PROGRAMSKI JEZIKI IN PROGRAMIRANJE
NALOGE S KRATKIMI ODGOVORI

17. Kako imenujemo program, ki prevede programsko kodo iz višjega programskega jezika (Pascal ali C) v strojno kodo?

(2 točki)

Rešitev:

prevajalnik ali interpreter (tolmač)

18. Na začetku programa so dane naslednje deklaracije:

Pascal:

```
var x,y,t : integer;
```

C:

```
int x,y,z;
```

Napišite zaporedje stavkov, ki zamenja vsebini spremenljivk x in y .

(2 točki)

Rešitev:

Pascal:

```
t:=x;  
x:=y;  
y:=t;
```

C:

```
t=x;  
x=y;  
y=t;
```

19. Napišite sled izpisa naslednjega programa. Pazite na presledke, ki jih označite z veliko črko B. Pazite tudi na skoke v novo vrsto.

(2 točki)

Pascal:

```
var i,j: integer;
begin
  for i:=1 to 3 do begin
    j:=10-i;
    while j>2*i do begin
      writeln('SA':j);
      dec(j);
    end;
  end;
end.
```

C:

```
#include <stdio.h>
#include <stdlib.h>
#include <string.h>
void main()
{
  char obl[5]="%", dol[2];
  for (int i=1;i<=3;i++){
    int j=10-i;
    while (j>2*i){
      itoa(j,dol,10);
      strcpy(obl,"%");
      strcat(obl,dol);
      strcat(obl,"s\n");
      printf(obl,"sa");
      j--;
    }
  }
}
```

Rešitev:

```
      SA
     SA
    SA
   SA
  SA
 SA
SA
      SA
     SA
    SA
   SA
  SA
 SA
```

20. Ugotovite izpis naslednjega programa:

(prvi dve vrstici izpisa 1 točka, vse tri vrstice izpisa 2 točki)

Pascal:

```
var a,b,c:integer;
procedure kaj(a,b:integer;var c:integer);
begin
    a:=a+b;
    b:=sqr(b);
    c:=10+a;
    writeln(a:3,b:3,c:3);
end;
begin
    a:=7; b:=2; c:=15;
    writeln(a:3,b:3,c:3);
    kaj(c,a,b);
    writeln(a:3,b:3,c:3);
end.
```

C:

```
#include <stdio.h>
int a,b,c;
void kaj(int a, int b, int *c){
    a+=b;
    b*=b;
    c=10+a;
    printf("%3d %3d %3d \n",a,b,c);
}

void main() {
    a=7;b=2;c=15;
    printf("%3d %3d %3d \n",a,b,c);
    kaj(c,a,&b);
    printf("%3d %3d %3d \n",a,b,c);
}
```

Rešitev:

```
7  2 15
22 49 32
7 32 15
```

SKLOP PROGRAMSKI JEZIKI IN PROGRAMIRANJE

STRUKTURIRANE NALOGE

21. V datoteki `uspeh.dat` so za vsakega dijaka zapisani naslednji podatki: ime (10 znakov), priimek (20 znakov), razred (3 znaki) in uspeh (celo število od 1 do 5).

Deklarirajte zapis in datoteko.

Napišite program, ki na osnovi podatkov v datoteki `uspeh.dat` v enodimenzijsko tabelo realnih števil zapiše odstotke nezadostnih, zadostnih, dobrih, prav dobrih in odličnih dijakov.

*(deklaracija podatkovnih tipov 1 točka
program 3 točke)*

Rešitev:

Pascal:

```
type dijak = record
    ime : string[10];
    priimek : string[20];
    razred : string[3];
    uspeh : integer;
end;
datoteka = file of dijak;
tabela = array[1..5] of real;
var f : datoteka;
    x : dijak;
    i,st_rec : integer;
    odstotek : tabela;
begin
    for i:=1 to 5 do
        odstotek[i]:=0;
    assign(f,'uspeh.dat');
    {$I-}
    reset(f);
    if IOResult<>0 then
        writeln('Napaka pri odpiranju datoteke uspeh.dat!')
    else begin
        {$I+}
        st_rec:=0;
        while (not(eof(f))) do begin
            read(f,x);
            odstotek[x.uspeh]:=odstotek[x.uspeh]+1;
            inc(st_rec);
        end;
        if (st_rec=0) then
            writeln('Datoteka je prazna')
        else
            for i:=1 to 5 do
                odstotek[i]:=odstotek[i]/st_rec;
            close(f);
        end;
    end.
end.
```

C:

```
#include <stdio.h>
struct dijak {
    char ime[10];
    char priimek[20];
    char razred[3];
    int uspeh;
} x;
int main()
{
    FILE *f;
    float odstotki[5]={0};
    int i, st_rec=0;
    if ((f=fopen("uspeh.dat", "r"))==NULL)
        printf("Napaka pri odpiranju datoteke uspeh.dat!!!");
    else {
        fread(&x, sizeof(x), 1, f);
        while (!feof(f)) {
            odstotki[x.uspeh-1]++;
            st_rec++;
            fread(&x, sizeof(x), 1, f);
        }
        if (st_rec==0) {
            printf("Datoteka je prazna");
        }
        else {
            for (int i=0; i<5; i++) {
                odstotki[i]/=st_rec;
            }
        }
        fclose(f);
    }
}
```

22. Napišite program, ki najprej prebere število vrstic n , nato pa z gnezdenjem zank generira izpis števil v trikotni obliki tako, kakor prikazujeta naslednja primera:

a) $n=5$	1 23 345 4567 56789	b) $n=8$	1 23 345 4567 56789 678901 7890123 89012345
----------	---------------------------------	----------	--

(4 točke)

Rešitev:

Pascal:

```
var x,y,n:integer;

begin
  read(n);
  for x:=1 to n do begin
    for y:=n downto x+1 do
      write(' ');
    for y:=x to 2*x-1 do
      write (y mod 10);
    writeln;
  end;
end.
```

C:

```
#include <stdio.h>

void main()
{ int n,i,j,a;
  printf("vpisi stevilo vrstic");
  scanf("%d",&n);
  for (i=1;i<=n;i++) {
    for (j=n;j>=i+1;j--)
      printf(" ");
    a=i;
    for (j=1;j<=i;j++) {
      printf("%d",a % 10);
      a++;
    }
    printf("\n");
  }
}
```

23. Pri računanju amortizacije upoštevamo, da se vrednost osnovnega sredstva vsako leto zmanjša za kvocient med nabavno vrednostjo in življenjsko dobo. Če je npr. nabavna vrednost osnovnega sredstva 80000 SIT in življenjska doba 5 let, se vrednost osnovnega sredstva vsako leto zmanjša za 16000 SIT. Napišite podprogram, ki za vsako leto »življenja« osnovnega sredstva izpiše naslednje podatke:

- zaporedno številko leta,
- vrednost osnovnega sredstva na koncu tistega leta.

Nabavna vrednost osnovnega sredstva in življenjska doba naj bosta podani kot parametra. Za osnovno sredstvo z nabavno vrednostjo 80000 SIT in življenjsko dobo 5 let naj ima izpis naslednjo obliko:

```
konec leta 1   trenutna vrednost:   64000.00
konec leta 2   trenutna vrednost:   48000.00
konec leta 3   trenutna vrednost:   32000.00
konec leta 4   trenutna vrednost:   16000.00
konec leta 5   trenutna vrednost:     0.00
```

(4 točke)

Če je kandidat pravilno rešil nalogo, vendar je pri tem napisal le program, dobi 3 točke.

Rešitev:

Pascal:

```
procedure izpis(nabavna:real;ziv_doba:integer);
var tr_cena,faktor : real;
    i : integer;
begin
    tr_cena:=nabavna;
    faktor:=nabavna/ziv_doba;
    for i:=1 to ziv_doba do begin
        tr_cena:=tr_cena-faktor;
        writeln('konec leta ',i:2,' trenutna vrednost: ',tr_cena:10:2);
    end;
end;
begin
    izpis(80000,5);
end.
```

C:

```
#include <stdio.h>

void izpis(float nabavna, int ziv_doba)
{
    float trenutna_cena,faktor;
    trenutna_cena=nabavna;
    faktor=nabavna/ziv_doba;
    for (int i=1;i<=ziv_doba;i++) {
        trenutna_cena=trenutna_cena-faktor;
        printf("konec leta %2d trenutna vrednost: %10.2f\n",i,trenutna_cena);
    }
}

void main()
{
    izpis(80000,15);
}
```

24. Na neki cestninski postaji so teden dni spremljali pretok petih različnih razredov vozil. Preštete količine so vnesli v dvodimenzionalno tabelo `pretokVozil`, v kateri prva dimenzija predstavlja 7 dni v tednu, druga pa 5 razredov vozil.

Pascal:

```
type t_pretok=array[1..7,1..5] of integer;
var pretokVozil:t_pretok;
```

C:

```
int pretokVozil[7][5];
```

Napišite funkcijski podprogram `vsotaDan`, ki za izbrani dan v tednu vrne seštevek vseh vozil, ki so tisti dan peljala skozi cestninsko postajo. Predpostavite, da sta zaporedna številka dneva, ki predstavlja indeks ustrezne vrstice v tabeli `pretokVozil`, in tabela `pretokVozil` dani kot parametra.

Napišite funkcijski podprogram `maxDan`, ki s pomočjo podprograma `vsotaDan` (iz prvega dela naloge) poišče indeks tistega dne v tednu, ko je skozi cestninsko postajo peljalo največ vozil. Tabela `pretokVozil` naj bo dana kot parameter, predpostavite pa, da obstaja en sam dan z maksimalno vrednostjo.

(4 točke)

Rešitev:

Pascal:

```
function vsotaDan(dan:integer;data:t_pretok):longint;
var i : integer;
    vs : longint;
begin
    vs:=0;
    for i:=1 to 5 do
        vs:=vs+data[dan,i];
    vsotaDan:=vs;
end;
function maxDan(data:t_pretok):integer;
var d,i : integer;
begin
    d:=1;
    for i:=2 to 7 do
        if vsotaDan(i,data)>vsotaDan(d,data) then
            d:=i;
    maxDan:=d;
end;
```


C:

```
#include <stdio.h>

long int vsotaDan(int dan, int data[7][5])
{ long int vs=0;
  for (int i=0;i<5;i++)
    vs+=data[dan][i];
  return(vs);
}
int maxDan(int data[7][5])
{ int d=0;
  for (int i=1;i<7;i++)
    if (vsotaDan(i,data)>vsotaDan(d,data))
      d=i;
  return(d+1);
}
```

IZPITNA POLA 2

Obkrožite pravilno trditev.

1. Kateri od navedenih protokolov pripada aplikacijski plasti?

(2 točki)

- A SMTP (Simple Mail Transfer Protocol)
- B UDP (User Datagram Protocol)
- C CSMA/CD (Carrier Sense Multiple Access / Collision Detection)
- D RJ 45 TIA/EIA 568A standard (Registered Jack - Telephone Industry Association/
Electronics Industry Association)
- E IP (Internet Protocol)

2. Prenos podatkov med aplikacijama na končnih računalnikih v internetnem omrežju enoumno določajo:

(2 točki)

- A IP naslov izvora in MAC naslov izvora, IP naslov cilja in MAC naslov cilja;
- B IP naslov izvora in naslov vrat (port-a) izvora, IP naslov cilja in naslov vrat (port-a) cilja;
- C IP naslov izvora, IP naslov cilja;
- D IP naslov izvora, IP naslov cilja, vrsta prenosnega medija;
- E IP naslov izvora, IP naslov cilja, pristopna metoda.

3. Protokol TCP se od protokola UDP razlikuje tudi po tem, da:

(2 točki)

- A zagotavlja hitrejši prenos;
- B je protokol, ki je primeren le za prenose v realnem času;
- C zagotavlja zanesljivejši prenos med končnima računalnikoma;
- D ga podpira več proizvajalcev;
- E je primeren le za prenos po bakrenih vodnikih.

4. Če poznamo IP naslov enega od računalnikov v podomrežju (subnet) in masko podomrežja, lahko določimo:

(2 točki)

- A samo razred omrežja;
- B samo velikost podomrežja;
- C samo naslov podomrežja;
- D ničesar, ker so ti podatki pomembni le za usmerjevalnike, prometa k računalniku;
- E razred omrežja, naslov podomrežja, velikost podomrežja in interval IP naslovov računalnikov v podomrežju.

5. Podatkovno neodvisnost podatkov v podatkovni bazi zagotavlja:

(2 točki)

- A upravitelj PB,
- B uporabniške aplikacije,
- C SQL,
- D SUPB,
- E operacijski sistem.

6. Logična datoteka je:

(2 točki)

- A množica logičnih zapisov enakega tipa;
- B urejena množica povezanih zapisov;
- C množica podatkovnih elementov enakega tipa;
- D množica opisov zapisov;
- E množica opisov podatkovnih elementov.

7. Izhodna funkcija informacijskega sistema zajema:*(2 točki)*

- A zajemanje in zbiranje vseh elementov, ki jih IS potrebuje za obdelavo;
- B preoblikovanje in obdelavo podatkov;
- C posredovanje podatkov in informacij končnim uporabnikom;
- D nadzor in vrednotenje izhoda za preverjanje doseganja ciljev IS;
- E vzdrževanje izhodnih naprav (tiskalnikov, monitorjev ...).

8. Za razvoj IS so odgovorni:*(2 točki)*

- A samo informatiki,
- B samo programerji,
- C samo sistemski analitiki,
- D samo uporabniki IS,
- E samo vodilni delavci,
- F vsi navedeni poklici.

9. Redki indeksi se lahko uporabljajo:*(2 točki)*

- A če je indeksirana datoteka urejena po vrednosti indeksnega elementa;
- B če je indeksirana datoteka redka;
- C če je indeksirana datoteka ni fragmentirana;
- D če so zapisi indeksirane datoteke razpršeni;
- E v vsakem primeru.

10. Podatkovne tokove, ki jih zasledimo na diagramih toka podatkov, opišemo v:*(2 točki)*

- A minispecifikacijah,
- B odločitvenih tabelah,
- C strukturnih diagramih,
- D podatkovnem slovarju.
- E entitetno-relacijskem modelu.

11. Določite vrstni red faz življenjskega cikla programske opreme. Poleg faze napišite zaporedno številko:

(2 točki)

Odgovor:

specifikacija zahtev ____ (1) ____

implementiranje ____ (4) ____

vzdrževanje ____ (7) ____

prenos v ciljno okolje ____ (6) ____

testiranje ____ (5) ____

načrtovanje ____ (3) ____

analiza zahtev ____ (2) ____

12. Kako imenujemo orodja, s katerim računalniško podpremo celoten cikel (razvoj in vzdrževanje) programske opreme?

(2 točki)

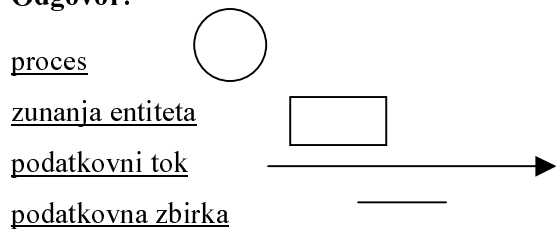
Odgovor:

Orodja CASE./Integrirano orodje CASE.

13. Narišite in poimenujte komponente diagramov toka podatkov (DTP).

(2 točki)

Odgovor:

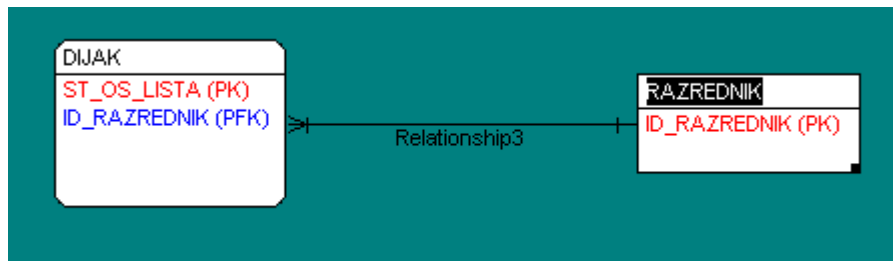


14. V obliki modela E-R predstavite entitetna tipa DIJAK in RAZREDNIK ter razmerje med njima.

(2 točki)

Če je kandidat pravilno zapisal entitetna tipa in je naredil napako pri števnosti razmerja, dobi 1 točko.

Odgovor:



15. Podatkovna baza je sestavljena iz naslednjih tabel:

Oddelek (Sifra_oddelka, Ime_oddelka)

Usluzbenec (Sifra_usluzbenca, Priimek, Ime, Placa, Sifra_oddelka)

Projekt(Sifra_projekta, Ime_projekta, Datum_zacetka, Datum_konca)

Dela(Sifra_projekta, Sifra_usluzbenca)

Napišite stavek SQL, ki vrne abecedno urejen seznam priimkov in imen uslužbencev, z vsaj 200.000 SIT plače.

(1 točka)

Rešitev:

```
SELECT U.Priimek, U.Ime
```

```
FROM Usluzbenec U
```

```
WHERE U.Placa >= 200000;
```

Napišite stavek SQL, ki za vsak oddelek izpiše število projektov, pri katerih sodeluje.

(3 točke)

Rešitev:

```
SELECT O.Ime_oddelka, COUNT(distinct D.Sifra_projekta)
```

```
FROM Oddelek O, Usluzbenec U, Dela D
```

```
WHERE (O.Sifra_oddelka=U.Sifra_oddelka) and
```

```
(U.Sifra_usluzbenca=D.Sifra_usluzbenca)
```

```
GROUP BY O.Ime_oddelka;
```

16. Podan je kontekstni diagram za spletni informacijski sistem, ki bo omogočal obdelavo prijav za vpis na univerzo. Sistem bodo lahko uporabljali dijaki, vpisna služba na univerzi in posamezne fakultete. Dijaki bodo lahko v elektronski obliki oddali svojo prijavo za vpis, vpisna služba pa bo potem vnesla podatke o uspehu v 3. in 4. letniku srednje šole ter število točk, doseženih pri maturi. Na podlagi teh podatkov bo vpisna služba razvrstila kandidate, tako izračunan vrstni red pa bo omogočil izdelavo seznamov tistih kandidatov, ki se lahko vpišejo v posamezne študijske programe (kar je še zlasti pomembno pri omejitvi vpisa). Ti seznamji bodo v elektronski obliki dostopni fakultetam, vsak dijak pa bo dobil obvestilo, ali je sprejet ali ne.

Narišite prvi nivo diagrama podatkovnih tokov, iz katerega naj bodo razvidni najpomembnejši procesi in podatkovne zbirke.

(4 točke)

Če je kandidat pravilno zapisal in povezal zunanje entitete s procesi, dobi 1 točko.
Če je kandidat pravilno zapisal in povezal zbirko podatkov s procesi, dobi še 1 točko.

Rešitev:

