

Име и презиме ученика _____

ПРИМЕЊЕНЕ ИНФОРМАЦИОНЕ ТЕХНОЛОГИЈЕ - 4

ИНИЦИЈАЛНИ ТЕСТ

(ово је као **МАТУРСКИ ИСПИТ**)

(**ИСПИТ ЗА ПРОВЕРУ СТРУЧНО –ТЕОРИЈСКИХ ЗНАЊА**)

СКАЛА ЗА ПРЕВОЂЕЊЕ БОДОВА У УСПЕХ	
Постигнут број бодова	Успех

до 50	недовољан (1)
50,5 – 63	довољан (2)
63,5 – 75	добар (3)
75,5 – 87	врло добар (4)
87,5 - 100	одличан (5)

ПОСТИГНУТ БРОЈ
БОДОВА

/100

ОЦЕНА

УПУТСТВО ЗА РАД

- ❖ Тест који треба да решите има 50 задатака. За рад је предвиђено 120 минута.
- ❖ Сваки задатак пажљиво прочитајте, размислите о томе шта се у задатку тражи и пратите упутства о начину на који треба да га решите (заокруживање, повезивање, дописивање и друго).
- ❖ Задатке решавајте хемијском оловком. Током рада можете да користите графитну оловку и гумицу. Пре него што предате тест, проверите своје одговоре још једном, а затим оне одговоре који су написани графитном оловком подебљајте хемијском. Одговор који је написан само графитном оловком неће бити признат, као ни одговор који је прецртан или исправљен хемијском оловком. Уколико се у задатку тражи и приказ поступка рада, потребно је да и њега испишете хемијском оловком.
- ❖ Важно је да пажљиво одговарате на питања јер **сваки нетачан одговор (део одговора) доноси 0 бодова** за задатак **у целини**.

НАПОМЕНЕ У ВЕЗИ ОЦЕЊИВАЊА ИНИЦИЈАЛНОГ ТЕСТА:

Оцена се рачуна по шеми са прве стране, али у дневник не улази бројчано

Оцена у дневник улази као активност (смешак) :

зелени за добијену позитивну оцелу
жути ако је и поред уложеног труда и даље оцелу 1
црвени ... све остало

Постоји могућност и уписивања осталих повратних информација
(степен савладаности градива из претходних школских година,
шта је потребно поново проћи,)

Уписује се (као коментар) и која би то била оцелу да је матура у питању !

У следећим задацима заокружите број испред траженог одговора

1. Заокружити исказ који описује улогу Domain Name Server-a:

	/	1
--	---	---

1. превођење имена домена у IP адресу
2. хостовање веб сајта
3. вршење функције главног чвора у локалној рачунарској мрежи
4. приказ динамичких веб страница

2. Међу понуђеним ентитетима, одредити ентитет са атрибутима који **НИСУ** одговарајући. Заокружити број испред траженог одговора:

	/	1
--	---	---

1. Ентитет: СТУДЕНТ – Атрибути: име, презиме, смер, број бодова, просек
2. Ентитет: ДРЖАВА – Атрибути: назив, број становника, површина
3. Ентитет: КЊИГА – Атрибути: наслов, аутор, година издања, издавач, адреса издавача, телефон издавача
4. Ентитет: АВИОН – Атрибути: произвођач, марка, година производње, број седишта

3. Заокружите тачан исказ:

	/	1
--	---	---

1. Примарни кључ је атрибут који мора бити целобројног типа.
2. Примарни кључ је атрибут који указује на зависност од неке друге табеле.
3. Примарни кључ је атрибут који јединствено идентификује врсте у табели.
4. Ако табела садржи вишевердносни атрибут, њему се додељује функција примарног кључа.

4. Дат је упит:

	/	1
--	---	---

SELECT *

FROM ucenici

WHERE odeljenje=4 OR odeljenje=7 OR odeljenje=10

Заокружити оператор који треба користити у датом упиту да би се избегло вишеструко коришћење оператора OR:

1. LIKE
2. BETWEEN
3. AND
4. IN

-
5. Заокруживањем редног броја обележити клаузулу коју је потребно користи уколико листа иза резервисане речи SELECT садржи агрегатну функцију и једну или више колона које нису део агрегатне функције:

	/	1
--	---	---

1. HAVING клаузулу
 2. GROUP BY клаузулу
 3. JOIN клаузулу
 4. ORDER BY клаузулу
-

6. Заокружити број испред исказа који представља исправан наставак дате тврдње:

	/	1
--	---	---

При креирању објеката изведене класе...

1. извршава се само конструктор изведене класе
 2. прво се извршава конструктор родитељске класе, али само ако је позван кључном речју **base**
 3. обавезно се прво извршава конструктор изведене, а потом конструктор родитељске класе
 4. обавезно се прво извршава конструктор родитељске, а потом конструктор изведене класе
-

7. У програму написаном у програмском језику Ц декларисана је променљива `fp` која представља показивач на бинарну датотеку и променљива **podatak** чија вредност ће се уписати у дефинисану бинарну датотеку.

	/	1
--	---	---

Заокружити редни број испред наредбе која омогућава упис три бајта у бинарну датотеку:

1. `fprintf(&podatak, 3, 1, fp);`
 2. `fprintf(fp, 3, &podatak);`
 3. `fwrite(podatak, 24, 1, fp);`
 4. `fwrite(&podatak, 24, 0, fp);`
 5. `fwrite(&podatak, 3, 1, fp);`
-

8. Заокружити број испред исказа који тачно дефинише појам веб сајта:

	/	1
--	---	---

1. скуп протокола за отпремање и преузимање података са интернета, као и протоколи за комуникацију на интернету
 2. скуп веб-страница које могу да садрже текст, слике, видео снимке и други мултимедијални садржај састављен у једну целину
 3. веб сајт чини интернет прегледач - програм који се користи за читање хипертекст докумената
 4. интернет сервис (`www`) који омогућава корисницима размену докумената која се састоје од текста, слика и мултимедијалних компоненти
-
-

9. Извршава се следећа SELECT наредба:

	/	2
--	---	---

```
SELECT MIN(Datum_Zaposlenja), Odsek_Id
FROM Zaposleni
GROUP BY Odsek_Id
```

Заокруживањем броја испред одговарајућег исказа, одредити које ће вредности бити приказане:

1. Најранији датум запослења за сваки одсек предузећа.
2. Најранији датум запослења у целом предузећу.
3. Датум запослења последњег запосленог радника у целом предузећу.
4. Датум запослења последњег запосленог радника за сваки одсек.
5. Датум запослења најстаријег запосленог радника у сваком одсеку предузећа.

10. Извршава се упит:

	/	2
--	---	---

```
SELECT prezime, ime, email
FROM ucenik
ORDER BY prezime
WHERE prosek >= 4.50
```

Наредба се неће извршити. Заокружити број испред разлога услед кога се наредба неће извршити:

1. Наредба се неће извршити једино ако нема ни једног одличног ученика.
2. Услов треба написати у HAVING клаузули
3. Потребно је назначити редослед сортирања (asc, desc).
4. Потребно је променити редослед клаузула.

11. У програмском језику Ц је дата декларација променљивих, а касније у коду извршен позив функције на следећи начин:

	/	2
--	---	---

```
int x, y, i, j;
float **mat, *vek, z;
mat[i] = Formiraj(x, 0.5);
```

На основу позива, проценити каквог је облика прототип функције и заокружити број испред тачно написаног прототипа:

1. float Formiraj(int n, float m);
 2. void *Formiraj(int n, int m);
 3. float *Formiraj(float n, int m);
 4. float *Formiraj(int n, float m);
 5. float **Formiraj(int n, float m);
-
-

12. Дат је део кода на програмском језику Ц:

	/	2
--	---	---

```
for(j=0; j<n; j++)  
    if(a[j]>0) s+=a[j];  
    else break;
```

Свака for петља може се написати коришћењем while и do-while наредбе. Заокружити број испред понуђеног кода који је еквивалентан коду датом у тексту задатка:

1. j=0;
while(j<n && a[j]>0) s+=a[j++];
2. j=0;
while(j<n && a[j++]>0) s+=a[j];
3. j=0;
while(j<n || a[j]>0) s+=a[j++];
4. j=0;
while(j<n && a[j]<=0) s+=a[j++];

13. У програмском језику С# дата је декларација једне стринг и једне целобројне променљиве, као и део кода:

	/	2
--	---	---

```
string str = "Primer";  
int broj = 66;  
Console.WriteLine(str + broj + 65);  
Console.WriteLine(broj + 65 + str);
```

Анализирати код и проценити шта ће се приказати на екрану након његовог извршења. Заокружити број испред очекиваног одговора:

- | | | | |
|---------------|---------------|--------------|-------------|
| 1. Primer6665 | 2. Primer6665 | 3. Primer131 | 4. PrimerBA |
| 131Primer | 6665Primer | 131Primer | BAPrimer |

14. Заокружити број испред исказа који представља исправан наставак дате реченице:

	/	2
--	---	---

Ако try-catch наредба има више catch блокова у којима "хватамо" изузетак основне **Exception** класе, заједно са изузецима других класа изведених из класе **Exception**...

1. онда се изузетак основне Exception класе може „хватати“ у било ком catch блоку (редослед није битан, битно је да се наведу све могуће грешке)
2. онда се изузетак основне Exception класе мора „хватати“ у последњем catch блоку
3. онда се изузетак основне Exception класе мора „хватати“ у првом catch блоку
4. основна Exception класа се не комбинује у истој наредби са класама изведеним из ње јер их основна класа „маскира“

15. У програмском језику C# дат је рекурзивни метод који проверава да ли је неки стринг палиндром. Да би код био комплетиран потребно је допунити трећи ред условом `if` наредбе.

	/	2
--	---	---

```
1. public static bool palindrom(String s)
2. {
3.     if (s.Length <= 1) return true; //bazni slučaj
4.     else if (_____) return false;
5.     else return palindrom(s.Substring(1, s.Length - 2));
6. }
```

Заокружити број испред очекиваног одговора:

1. `s[0] != s[s.Length - 1]`
2. `s[0] != s[s.Length]`
3. `s[1] != s[s.Length - 1]`
4. `s[1] != s[s.Length]`

16. Дат је код програма у програмском језику C# којим су дефинисане две класе: `class Program` која садржи `Main(string[] args)` методу и `class KlasaA`. Анализирати дати код и одредити да ли је код исправно написан. Понуђени одговори дају опис последица извршавања овог кода. Заокружити број испред тачног исказа.

	/	2
--	---	---

```
class Program {
    public static void Main(string[] args) {
        Object a1 = new KlasaA();
        Object a2 = new KlasaA();
        Console.WriteLine(a1.Equals(a2));
    }
}

class KlasaA {
    int x;
    public bool Equals(KlasaA a) {
        return this.x == a.x;
    }
}
```

1. Програм има грешку, јер се изразом `a1.Equals(a2)` проверава једнакост објеката `a1` и `a2` различитог типа од `Object`.
2. Програм има грешку јер се једнакост објеката `a1` и `a2` типа `KlasaA` проверава изразом `a1 == a2`.
3. Програм се извршава без грешке и приказује се `true` на екрану.
4. Програм се извршава без грешке и приказује се `false` на екрану.

-
17. Дата је дефиниција класе у програмском језику C#. Проценити где у дефиницији класе (испред које методе) треба заменити знакове ??? службеном речју **static**.

	/	2
--	---	---

```
1. public class Test {
2.     private int broj;
3.
4.     public ??? int kvadrant(int n) { return n * n; }
5.     public ??? int getBroj() { return broj; }
6. }
```

Заокружити број испред тачне изјаве:

1. Метода **kvadrant** МОРА да буде статичка, док метода **getBroj** може и не мора.
2. Обе методе морају бити статичке.
3. Ни једна од дефинисаних метода није статичка.
4. Метода **getBroj** НЕ СМЕ да буде статичка, док метода **kvadrant** може и не мора.

-
18. Дат је код у програмском језику C#, који дефинише рекурзивни метод. Анализирати код и одредити резултат који ће се приказати на екрану.

	/	2
--	---	---

```
namespace TestPrimer {
    class Program {
        static void Main(string[] args) {
            fun(2);
        }
        public static void fun(int n) {
            while (n > 1) {
                Console.WriteLine((n - 1) + " ");
                fun(n - 1);
            }
        }
    }
}
```

Заокружити број испред очекиваног одговора:

1. Програм на екрану не приказује ништа
 2. Програм на екрану приказује 1 2 3
 3. Програм на екрану приказује 3 2 1.
 4. Програм на екрану бесконачно приказује 1 1 1 1
 5. Програм на екрану бесконачно приказује 2 2 2 2
-
-

У следећим задацима заокружите бројеве испред тражених одговора

19. Заокружити бројеве испред **ТАЧНИХ** исказа који се односе на дефиницију **while** циклуса у програмском језику Ц:
- | | | |
|--|---|---|
| | / | 1 |
|--|---|---|
- код while циклуса се може десити да се тело циклуса не изврши ниједном (на почетку услов није задовољен),
 - while циклус се користи када се зна колико ће се пута циклус извршавати,
 - while циклус се извршава све док је услов логичка неистина (једнак нули),
 - у while циклусу се увек прво проверава да ли је услов логичка истина, те ако јесте наредба се извршава.

20. Обележити команде које се сматрају командама ажурирања података у бази података. Заокружити бројеве испред тражених одговора:
- | | | |
|--|---|-----|
| | / | 1,5 |
|--|---|-----|
- Организовање податка
 - Додавање нових података
 - Брисање старих података
 - Враћање оштећених података у коректно стање
 - Измена постојећих података
 - Додела права приступа подацима
 - Измена структуре постојећих табела у бази

21. Заокружити бројеве испред наредби које служе за креирање, брисање и измену структуре релационе базе и објеката који чине релациону базу:
- | | | |
|--|---|-----|
| | / | 1,5 |
|--|---|-----|
- ALTER TABLE
 - INSERT
 - CREATE TABLE
 - DROP TABLE
 - UPDATE
 - DELETE TABLE
 - ADD COLUMN
 - ADD CONSTRAINT

-
22. Дате су наредбе декларације променљивих (са и без иницијализације вредности) написане на програмском језику Ц. Заокружити бројеве испред исправно написаних наредби декларације променљивих:

	/	1,5
--	---	-----

1. `char zn="a";`
2. `char zn='\b';`
3. `int a=b=c=5;;`
4. `int a=5, b=5, c=5;`
5. `int a=0xf2;`
6. `long a; b=5; c;`

-
23. Дате су наредбе декларације и иницијализације једнодомениционалног низа целих бројева у програмском језику Ц. Заокружити бројеве испред исправно написаних наредби декларације и иницијализације једнодимензионалног низа:

	/	1,5
--	---	-----

1. `int a[10]={1,2,3};`
2. `int a[5]={-3, -2, -1, 0, 1, 2, 3};`
3. `int a[]={10,20,30,40,50};`
4. `int[5] a={1, 2, 3, 4, 5};`
5. `int a={10,20,30,40,50};`
6. `int a[5]={ '1', '2', '3', '4', '5' };`

-
24. Заокружити бројеве испред понуђених тврдњи које представљају тачне наставке изјаве која се односе на повратну вредност функције **fopen** у програмском језику Ц:

	/	2
--	---	---

При покушају да датотеку отворимо за писање, функција **fopen**...

1. ако датотека не постоји, креира празну датотеку, поставља се на њен почетак и враћа показивач на ту датотеку
 2. враћа NULL показивач ако датотека не постоји
 3. ако датотека постоји, излази упозорење да ће њен садржај бити уништен при отварању
 4. ако датотека не постоји, изазива грешку која доводи до пуцања програма
 5. ако датотека постоји, уништава њен садржај без упозорења
-
-

-
25. У програму на програмском језику Ц извршена је следећа декларација, а касније и резервација меморијског простора за низ реалних бројева обичне талности дужине n:

	/	2
--	---	---

```
float *B;  
int n;  
B=(float*) calloc (n, sizeof (float));
```

Заокружити бројеве испред исправно написаних наредби за **УНОС** i-тог елемента низа B:

1. `scanf("%f", B+i);`
2. `scanf("%p", B+i);`
3. `scanf("%f", *(B+i));`
4. `scanf("%f", B[i]);`
5. `scanf("%f", &B[i]);`

-
26. Дат је код програма у програмском језику C#. Код садржи објекте две класе у којима је дефинисан метод **ToString()**. Анализирати код датог програма и одредити који од датих исказа су тачни.

	/	2
--	---	---

```
namespace TestPrimer {  
    class Program {  
        static void Main(string[] args) {  
            Object a = new Klasa();  
            Object obj = new Object();  
            Console.WriteLine(a);  
            Console.WriteLine(obj);  
        }  
    }  
}  
class Klasa{  
    int x;  
    public override string ToString() {return "x u A je " + x;}  
}
```

Заокружити бројеве испред очекиваних одговора:

1. Програм има грешку, јер наредбу **Console.WriteLine(a)** треба заменити наредбом **Console.WriteLine(a.ToString())**.
 2. Приликом извршавања наредбе **Console.WriteLine(a)**, програм позива метод ToString() наслеђен из класе Object.
 3. Приликом извршавања наредбе **Console.WriteLine(a)**, програм позива метод ToString() из класе Klasa.
 4. Приликом извршавања наредбе **Console.WriteLine(obj)**, програм позива метод ToString() из класе Object.
-
-

27. Дати су искази који у програмском језику C# дефинишу конструктор.

	/	2
--	---	---

Заокружити бројеве испред очекиваних одговора:

1. Подразумевани конструктор без параметара се увек аутоматски додаје класи.
2. Подразумевани конструктор без параметара се класи аутоматски додаје уколико у њој није експлицитно дефинисан ниједан конструктор.
3. У класи се мора експлицитно дефинисати бар један конструктор.
4. Конструктори немају тип резултата, чак ни **void**.

28. Заокружите тачне исказе. Интернет је:

	/	2
--	---	---

1. Интернет је светски систем умрежених рачунарских мрежа
2. Софтвер за преглед и приказ www страница се сматра Интернетом
3. Подаци који „путују“ светском мрежом и скуп корисника заједно чине Интернет мрежу
4. Интернет чини њена хардверска компонента као и систем софтверских слојева који контролишу различите аспекте њене комуникационе инфраструктуре

29. Дата је табела **Kupci** са структуром:

	/	2
--	---	---

```
( Id int primary key, Prezime varchar(50), Adresa varchar(50),  
Mesto varchar(20), Telefon varchar(5), Status varchar(8) )
```

И табела **NoviKupci** са структуром:

```
( Id int primary key, Prezime varchar(50), Telefon varchar(20), Status varchar(8) )
```

Извршава се упит:

```
INSERT INTO NoviKupci  
SELECT * FROM Kupci WHERE Status <> 'Aktivan'
```

Одредити шта је резултат извршења датог упита. Заокружити бројеве испред тражених одговора:

1. Како табела **NoviKupci** има све колоне које постоје и у табели **Kupci**, упит се извршава без грешке и у табелу **NoviKupci** се уписују записи из табеле **Kupci** са статусом који није **Aktivan**
 2. Упит се не извршава, пријављује грешку јер се број колона у табели **Kupci** разликује од броја колона у табели **NoviKupci**
 3. Упит би се извршио без грешке да су у SELECT клаузули подупита, уместо * наведене све колоне табеле **Kupci** које имају своју одговарајућу колону у табели **NoviKupci**
 4. Упит јавља грешку због покушаја уписа вредности у поље примарног кључа који је аутоматски, тј креира га сама база
-
-

30. Одредити ентитете који садрже одговарајуће атрибуте. Заокружити бројеве испред тражених одговора:

	/	3
--	---	---

1. Ентитет: КЊИГА – Атрибути: наслов, аутор, година издања, издавач, адреса издавача, телефон издавача
2. Ентитет: СТУДЕНТ – Атрибути: име, презиме, смер, број бодова, просек
3. Ентитет: ТУРИСТИЧКА АГЕНЦИЈА – Атрибути: назив, адреса, година оснивања, власник, стручна квалификација власника, запослени, стручна квалификација запослених
4. Ентитет: АВИОН – Атрибути: произвођач, марка, година производње, број седишта
5. Ентитет: ДРЖАВА – Атрибути: назив, број становника, површина
6. Ентитет: САЈАМ – Атрибути: назив, број излагача, покровитељ, адреса покровитеља, контакт особа покровитеља

31. Обележити ентитете код којих је извршен адекватан избор јединственог идентификатора. Заокружи бројеве испред тражених одговора:

	/	3
--	---	---

1. јединствени матични број грађанина (ЈМБГ) за ентитет ОСОБА
2. датум рођења за ентитет ОСОБА
3. ИСБН број за ентитет КЊИГА
4. регистарска ознака за АУТОМОБИЛ
5. дестинација за ентитет АРАНЖМАН
6. режисер за ентитет ФИЛМ

32. Одредити тачан исказ о оператору ANY који се примењује са подупитом који враћа више вредности:

	/	3
--	---	---

1. Оператор ANY врши поређење са свим вредностима које враћа подупит и враћа TRUE ако било која од вредности подупита задовољава услов
 2. Оператор ANY врши поређење са свим вредностима које враћа подупит и враћа TRUE ако све вредности подупита задовољавају услов
 3. Оператор ANY може да се користи испред кључне речи DISTINCT.
 4. Услов =ANY(скуп вредности) је еквивалентан услову IN (скуп вредности)
 5. Оператору ANY може да претходи оператор LIKE или оператор IN.
 6. Оператору ANY мора да претходи оператор поређења (=, <>, >, >=, <, <=)
-

33. Креирана је табела **SKOLA**, а затим су у њу уписани подаци извршавањем следећих наредби:

	/	3
--	---	---

```
create table Skola( skolaID int primary key, Naziv nvarchar(50) )
insert into Skola values (101, 'Nikola Tesla')
insert into Skola values (102, 'Mihajlo Pupin')
insert into Skola values (103, 'ETS Zemun')
```

За дати упит, треба проценити сценарио који ће се десити и заокруживањем редних бројева испред исказа, означити могуће исходе:

```
select * into StrucneSkole from Skola
```

1. Уколико се дода услов **where** 1=2 упит јавља грешку јер је 1=2 увек нетачно тј. **False**
2. Уколико се дода услов **where** 1=2 упит се извршава, креира се нова табела исте структуре као и табела Skola, али се у њу не уписује ни један ред
3. Уколико табела са именом StrucneSkole постоји у бази, упит јавља грешку
4. Уколико табела са именом StrucneSkole постоји у бази, креира се нова са именом StrucneSkole(1) и у њу се преписују сви редови из табеле Skola
5. Уколико табела са именом StrucneSkole постоји у бази, не креира се нова, само се у постојећу преписују редови из табеле Skola
6. Креира се копија табеле Skola - нова табела под именом StrucneSkole исте структуре као и табела Skola и у њу се преписују сви подаци из табеле Skola
7. Упит јавља грешку јер се кључна реч **into** користи искључиво у комбинацији са **insert**

34. Дат је прототип функције написан у програмском језику Ц:

	/	3
--	---	---

```
void Saberi(int n, int *a, int *b);
```

У main функцији дате су следеће декларације променљивих:

```
int x[50][50], y[50], m, j, i;
```

Заокружити бројеве испред исправно написаних позива декларисане функције:

1. Saberi(10, y, x[0]);
2. Saberi(y, x[i], x[i+1]);
3. Saberi(y[i], x[i], x[i+1]);
4. Saberi(x[i][j], x[i], x[j]);
5. Saberi(m, y[i], y[i+1]);
6. Saberi(m, y, x[i][j]);

35. Заокружити могуће наћине укључивања CSS стила у оквиру HTML стране:

	/	3
--	---	---

1. Увежена екстерна CSS датотека преко хипервезе у заглављу HTML веб странице - External Style Sheet
2. Дефинисан екстерни CSS стил приказивања у прегледачу HTML страница – CSS plugin
3. Дефинисан интерни CSS стил у оквиру head секције `<style type="text/css">...</style>` - Internal Style Sheet
4. Дефинисан стил у оквиру елемента HTML стране - Inline style
5. Дефинисан стил на крају HTML стране изван head и body секције – Outline style
6. Дефинисан CSS стил у оквиру секције body, унутар тара `<css></css>`

36. Повратне вредности акције MVC контролера (controller action method) могу бити различитих типова. Заокружити бројеве испред могућих повратних вредности.

3	/	3
---	---	---

1. ASPResult
2. ViewResult
3. JsonResult
4. MVCResult
5. RedirectResult
6. ModelResult

ОВАЈ ЗАДАТАК СЕ НЕ РАДИ У ИНИЦИЈАЛНОМ ТЕСТУ
јер је у питању градиво из 4. разреда
(Сва остала питања су из прва три разреда !)

Допуните следеће реченице и табеле

37. Допунити реченицу наводећи назив нормалне форме:

	/	1
--	---	---

Уколико сви атрибути релације који нису део кључа зависе од сваког атрибута који је део кључа кажемо да је релација у _____ нормалној форми.

38. Наредбама програмског језика Ц декларисана је правоугаона матрица и три целобројне променљиве:

	/	2
--	---	---

```
int mat[10][20]; int k, N, M;
```

где **N** представља број врста, а **M** број колона правоугаоне матрице **mat**.

Допунити изразима који недостају код петље која има задатак да дуплира све елементе прве врсте матрице:

```
for (k=0; k< _____; k++)
```

```
mat[_____][_____]*=2;
```

39. Дата су следеће декларације у програмском језику Ц: `int p[200], i, n, k;`

	/	2
--	---	---

А затим и део кода који треба да из низа **p** дужине **n**, сажимањем **ИЗБАЦИ** елеменат низа са позиције **k**, а затим ажурира нову дужину низа.

Имајући у виду дату иницијализацију петље, у предвиђена поља унеси одговарајуће елементе **преписивањем израза** из листе понуђених израза (подразумевати да су све потребне променљиве иницијализоване):

```
for(i=k; i _____; _____)
    _____ = _____;
n--;
```

1. `p[i+1]`
2. `p[i-1]`
3. `p[i]`
4. `p[k]`
5. `i++`
6. `i--`
7. `< n`
8. `< n-1`

40. Дат је део кода написан на програмском језику Ц:

	/	3
--	---	---

```
int a[7]={81,12,35,97,40,52,17}, *pa, x, y;
pa=a+3;
x = *(pa-2)+1;
y = (*pa-2)+1;
```

Анализирати код и одредити вредности променљивих **x** и **y**, као и показивача **pa**, по извршењу све три извршне наредбе датог кода:

`x = _____`

`y = _____`

`pa = a + _____`

41. На програмском језику Ц декларисан је и иницијализован стринг и два показивача:

	/	3
--	---	---

```
char s1[]="Short Message Service", *s2, *s3;
```

Одредити и на предвиђену линију уписати садржај означених стрингова по извршењу следећих наредби:

```
s2=strchr(s1, 'M') ;
s3=strrchr(s2, 'S') ;
strncpy(s1+1,s2,1) ;
strcpy(s1+2,s3) ;
```

`s1 = _____`

`s2 = _____`

`s3 = _____`

42. На програмском језику C# дефинисане су две класе:

	/	3
--	---	---

```
public class Racun {  
    public virtual int Uvecaj() { return 10; }  
}  
public class Dinarski: Racun {  
    public override int Uvecaj() { return 20 * base.Uvecaj(); }  
}  
public class Devizni : Racun {  
    public override int Uvecaj() { return 50 + base.Uvecaj(); }  
}
```

Унутар функције Main, креирана су три објекта ових класа на следећи начин:

```
Racun r = new Racun();  
Racun rDin = new Dinarski();  
Racun rDev = new Devizni();
```

Анализирати код и на предвиђене линије уписати шта метод Uvecaj() враћа при позиву из наведених објеката:

```
r.Uvecaj();  
_____  
rDin.Uvecaj();  
_____  
rDev.Uvecaj();  
_____
```

У следећим задацима повежите и уредите појмове према захтеву

43. Са леве страни дати су математички изрази, а са десне запис израза на програмском језику Ц. На линију испред записа израза, унети број којим је означен одговарајући израз:

	/	2
--	---	---

- $y = \frac{\sqrt{x} + 10}{a + |b|}$ _____ $y = \text{sqrt}(x+10) / (a+\text{fabs}(b))$
- $y = \frac{\sqrt{x} + 10}{a} + |b|$ _____ $y = \text{sqrt}(x)+10 / a+\text{fabs}(b)$
- $y = \frac{\sqrt{x} + 10}{a + |b|}$ _____ $y = \text{sqrt}(x+10) / a+\text{fabs}(b)$
- $y = \sqrt{x} + \frac{10}{a} + |b|$ _____ $y = (\text{sqrt}(x)+10) / (a+\text{fabs}(b))$

44. Дат је код на програмском језику Ц:

	/	2
--	---	---

```
switch (c) {  
    case 'A': case 'a': printf("Pravougaonik ");  
    case 'B': case 'b': printf("Trougao "); break;  
    case 'C': case 'c': printf("Krug ");  
    default: printf("Duz "); break;  
}
```

Са десне стране су дате вредности променљиве c (скретница), а са леве стране резулат извршења кода за дату вредност скретнице. На линију испред вредности скретнице унети редни под којим је наведен одговарајући екрански приказ:

- | | |
|----------------------------------|-----------|
| 1. Krug Duz | _____ 'b' |
| 2. Pravougaonik Trougao Krug Duz | _____ 'K' |
| 3. Krug | _____ 'A' |
| 4. Trougao | _____ 'c' |
| 5. Pravougaonik Trougao | |
| 6. Duz | |

45. Са леве стране су наведене наредбе програмског језика Ц за позиционирање у датотеци, а са десне описи ефеката датих наредби. На линију поред наредбе уписати редни број под којим је наведен опис ефекта наредбе:

	/	2
--	---	---

- | | | |
|-------------------------|-------|--|
| rewind(dat) | _____ | 1. одређује позицију у датотеци у виду броја бајтова од почетка датотеке |
| ftell(dat) | _____ | 2. позиционирање на почетак датотеке |
| fseek(dat, 0, SEEK_SET) | _____ | 3. позиционирање на крај датотеке |
| fseek(dat, 0, SEEK_END) | _____ | 4. ништа од понуђеног |

46. Са леве стране су дати нумерисани HTML тагови, а са десне су наведени индикатори. На линију уписати број којим се HTML таг повезује са одговарајућим индикатором који ће се видети на HTML страници:

	/	2
--	---	---

- | | | |
|---------------------------|-------|-----------------------|
| 1. <tr> | _____ | Дефинисање хиперлинка |
| 2. <td> | _____ | Нови ред у табели |
| 3. | _____ | Нова ћелија у реду |
| 4. | _____ | Уметање слике |

-
47. Са леве стране су дати нумерисани HTML тагови, а са десне су наведени индикатори. На линију уписати број којим се HTML таг повезује са одговарајућим индикатором који ће се видети на HTML страници:

	/	2
--	---	---

1. _____ приказ наслова трећег нивоа
2. <h3> _____ приказ текста дате величине
3. _____ приказ текста у боји
4. _____ приказ подебљаног текста

-
48. У JavaScript језику свака веб страница има објекте наведене у левој колони. На десној страни дати су описи наведених објеката. Повезати објекат са одговарајућим описом уносом редног броја којим је објекат нумерисан на означену линију:

	/	2
--	---	---

1. document _____ садржи URL адресе које су претходно биле посећене, као и URL адресе које су посећене након посете тренутној страници
2. history _____ садржи информације о URL адреси документа који је тренутно приказан у посматраном прозору
3. location _____ представља тренутно учитани документ и служи као приступна тачка садржају странице
4. window _____ објекат највишег нивоа који садржи све друге објекте и представља прозор претраживача

-
49. Са десне стране наведене су неке од функција библиотеке **ctype.h** програмског језика Ц, а са леве су дати њихови описи. Испред назива сваке од наведених функција, уписати редни број под којим је дат одговарајући опис:

	/	3
--	---	---

1. Да ли је с штампајући знак (укључујући и размак)? _____ isspace(c)
 2. Да ли је с велико слово? _____ isdigit(c)
 3. Да ли је с знак интерпункције? _____ isalpha(c)
 4. Да ли је с управљачки знак? _____ isupper(c)
 5. Да ли је с децимална цифра? _____ iscntrl(c)
 6. Да ли је с знак бели знак? _____ isprint(c)
 7. Да ли је с слово?
 8. Да ли је с хекса-децимална цифра?
-
-

50. Дата је табела **RADNIK**, табела **ODELJENJE** и упит:

	/	3
--	---	---

IDBR	IME	PREZIME	PLATA	BROD
5900	Slobodan	Golubović	900	10
5932	Mitar	Gavrilović	600	
5953	Persida	Kosanović	1100	20
6234	Marko	Pavlović	1300	30
6789	Janko	Nikolić	800	10

BROD	IMEOD	MESTO
50	Skladišta	Zemun
30	Marketing	Vračar
10	Plasman	Surčin
20	Direkcija	Grocka
40	Nabavka	Barajevo

Повезати упите и њихова значења уписом броја датог испред описа значења упита на одговарајућу линију:

`SELECT odeljenje.imeod,
radnik.prezime
FROM odeljenje LEFT JOIN radnik
ON radnik.brod = odeljenje.brod
WHERE radnik.brod IS NULL`

`SELECT odeljenje.imeod,
radnik.prezime
FROM odeljenje RIGHT JOIN radnik
ON radnik.brod = odeljenje.brod
WHERE odeljenje.brod IS NULL`

`SELECT odeljenje.imeod,
radnik.prezime
FROM odeljenje FULL JOIN radnik
ON radnik.brod = odeljenje.brod`

1. Приказује само раднике који нису распоређени у одељења
2. Приказује само одељења у којима нема радника
3. Приказује све раднике (и оне који јесу и оне који нису распоређени у одељења) и само она одељења у којима има радника
4. Приказује сва одељења - и она у којима има и оне у којима нема радника и све раднике – и оне који су распоређени у одељења, као и оне који нису распоређени