

Попуњава ученик:

Назив школе

Седиште школе

Образовни профил

Име и презиме ученика

Датум одржавања испита

Електротехничар информационих технологија – С#

МАТУРСКИ ИСПИТ

ИСПИТ ЗА ПРОВЕРУ СТРУЧНО –ТЕОРИЈСКИХ ЗНАЊА

Попуњава испитна комисија:

СКАЛА ЗА ПРЕВОЋЕЊЕ 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

Чланови испитне комисије:

Датум прегледа теста: _____

1. _____
2. _____
3. _____

УПУТСТВО ЗА РАД

- Тест који треба да решите садржи 100 бодова. 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**.
- Ако решавање задатака завршите раније, предајте тест и тихо изађите да се не би ометао рад других.

Желимо Вам много успеха у раду!

У следећим задацима заокружите број испред траженог одговора

1. ASP.NET MVC 3.0 долази са новом техником за дефинисање погледа (View Engine). Заокружити назив ове технике:

	/	1
--	---	---

1. ASP.NET View Engine
2. Salome
3. Razor
4. Default

2. Заокружити број испред тачног одговора. Који је од наведених начина за укључивање CSS стила најпогоднији за стилизовање, а посебно за касније одржавање и ажурирање HTML странице:

	/	2
--	---	---

1. Увежена екстерна CSS датотека преко хипервезе у заглављу HTML веб странице - External Style Sheet
2. Дефинисан интерни CSS стил у оквиру head секције `<style type="text/css">...</style>` - Internal Style Sheet
3. Дефинисан стил у оквиру елемента HTML стране - Inline style
4. Не постоји решење за укључивање CSS стила у стилизовање HTML странице

3. Заокружити број испред тачног одговора.

	/	1
--	---	---

Дефинисан је стил елемента у оквиру кога се налази позадинска слика. Коју вредност својство `background-repeat` треба да има уколико слика не треба да се понавља у оквиру елемента:

1. `background-repeat: fixed;`
2. `background-repeat: null;`
3. `background-repeat: no-repeat;`
4. `background-repeat: repeat-x;`

4. Заокружити од понуђених опција који симбол се користи за коментаре у ASP.NET MVC Razor синтакси:

	/	1
--	---	---

1. `//`
2. `/* ... */`
3. `<!--...-->`
4. `@* ... *@`

5. Дата је табела *PROJEKAT* над којом се извршава упит:

	/	2
--	---	---

`ALTER TABLE PROJEKAT ADD RokKraj date`

Одредити шта ће се десити након извршења упита. Заокружити број испред траженог одговора.

1. у табелу *PROJEKAT* додаје се ограничење *RokKraj*
2. у табелу *PROJEKAT* додаје се колона *RokKraj*
3. у табели *PROJEKAT* биће преименована колона
4. у базу података додаје се табела *PROJEKAT* са само једном колоном
5. у табели *PROJEKAT* промениће се тип података у колони *RokKraj*

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

	/	2
--	---	---

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

BROD	IMEOD	MESTO
50	Skladišt a	Zemun
30	Marketin g	Vračar
10	Plasman	Surčin
20	Direkcija	Grocka
40	Nabavka	Barajev o

`SELECT Imeod, AVG(Plata) AS ProsekPlata FROM Radnik, Odeljenje
WHERE Odeljenje.Brod=Radnik.Brod GROUP BY Imeod HAVING AVG(Plata)>1000`

Одредити резултат извршавања датог упита. Заокружити број испред траженог одговора:

1. Упит се не извршава зато што груписање мора да се изврши не само по називу одељења, него и по шифри одељења (*BrOd*)
2. Групишу по одељењима радници са платом већом од просечне плате
3. Приказују називи одељења и висина просечне плате у њима само за одељења у којима је просечна плата већа од 1000
4. Приказују називи одељења и висина просечне плате у њима, при чему се код одређивања просека узимају у обзир само плате веће од 1000

7. 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*

-
8. 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
-

9. У програмском језику Ц декларисана је целобројна променљива и додељена јој је вредност логичког израза:

	/	2
--	---	---

```
int x;  
x = 1==10>5;
```

Имајући у виду приоритет оператора, одредити вредност променљиве x после извршења ове наредбе. Заокружити број испред траженог одговора:

1. променљива добија вредност логичке неистине, тј. x = 0
 2. променљива добија вредност логичке истине, тј. x = 1
 3. вредност логичког израза се не може доделити целобројној променљивој
 4. променљива добија вредност логичке истине, тј. било који број различит од 0
-

10. Дата је декларација променљивих pod, br и део кода у програмском језику Ц:

	/	2
--	---	---

```
unsigned pod, br;  
pod=128;  
br=0;  
while(pod!=0) {  
    if(pod & 0x1) br++;  
    pod>>=0x1;  
}
```

Закључити шта представља вредност коју променљива br добије извршењем кода. Заокружити број испред траженог одговора:

1. Број јединица у бинарном запису броја pod
 2. Број нула у бинарном запису броја pod
 3. Број цифара у бинарном запису броја pod
 4. Број цифара у хексадецималном запису броја pod
-
-

11. У програмском језику Ц декларисане су променљиве и дат је део кода програма:

	/	1
--	---	---

```
FILE *fp;
char str[150];
fp=fopen("primer.txt", "r");
fgets(str, 80, fp);
```

Одредити шта је последица извршавања датог кода. Заокружити број испред очекиваног одговора:

1. Учитава 80 карактера из датотеке и смешта у стринг str.
2. Учитава максимално 150 карактера из датотеке и смешта у стринг str
3. Учитава стринг из датотеке све док се не прочита знак за нови ред или 80 карактера
4. Учитава стринг из датотеке све док се не прочита знак за нови ред или 150 карактера

12. У програмском језику Ц декларисани су структурни типови података **Tacka3D** (који дефинише тачку у простору) и **Lopta** (одређена центром и полупречником):

	/	1
--	---	---

```
typedef struct {
    float x, y, z;
}Tacka3D;

typedef struct {
    Tacka3D centar;
    float R;
}Lopta;
```

Заокружити број испред исправно написане наредбе декларације и иницијализације променљиве **x** типа **Lopta**, тако да јој центар буде у тачки $O(2,2,2)$, а полупречник 10цм:

1. `Lopta x={10, {2, 2, 2}};`
2. `Lopta x={2, 2, 2, 10};`
3. `Lopta x={2, 2, 2}, {10};`
4. `Lopta x={{2, 2, 2}, 10};`

13. Одредити која поља су заједничка и јединствена за све креиране објекте неке класе дефинисане у објектно оријентисаном програмском језику C#.

	/	1
--	---	---

Заокружити број испред очекиваног одговора:

1. Јавна
 2. Приватна
 3. Објектна
 4. Инстанцна
 5. Статичка
-
-

-
14. Дат је код програма у програмском језику C# који формира и штампа елементе низа **a**. Анализирати дати код и проценити шта ће се догодити након његовог извршавања.

	/	2
--	---	---

```
namespace TestPrimer {  
    class Program {  
        static void Main(string[] args){  
            int[] a = new int[5];  
            for (int i = 0; i < a.Length; i++) a[i] = i;  
            Console.Write(a[i] + " ");  
        }  
    }  
}
```

Заокружити број испред очекиваног одговора:

1. Програм приказује бројеве 0 1 2 3 4 на екрану.
2. Програм има грешку, јер ће у последњој наредби Console.Write метода Main покушати приступ непостојећем елементу a[5].
3. Програм приказује број 5 на екрану.
4. Програм има грешку, јер променљива **i** у последњој наредби Console.Write у методу Main неће имати дефинисану вредност.

-
15. Дат је део кода који је написан на C# програмском језику. Анализирати код и одредити шта ће се приказати на излазу.

	/	2
--	---	---

```
try  
{  
    int x = 0;  
    int y = 5 / x;  
}  
catch (Exception e)  
{  
    Console.WriteLine("Exception");  
}  
catch (ArithmeticException ae)  
{  
    Console.WriteLine(" Arithmetic Exception");  
}  
Console.WriteLine("finished");
```

Заокружи број испред тачног одговора:

1. Приказује се текст: finished
 2. Приказује се текст: Exception
 3. Ништа. Дешава се грешка приликом компајлирања
 4. Приказује се текст: Arithmetic Exception
-
-

16. Дат је код програма у програмском језику C# у ком су дефинисане три класе: `class Program` која садржи `Main(string[] args)` методу, `class A` и `class B`. Анализирати дати код и одредити да ли је код исправно написан.

	/	2
--	---	---

Заокружити број испред исказа који даје информацију о тачности кода.

```
class Program {
    public static void Main(string[] args) {
        B b = new B();
        b.Metod(5);
        Console.WriteLine("b.i je " + b.CitajI());
    }
}
class A {
    int i;
    public int CitajI(){return i;}
    public void Metod(int i) { this.i = i; }
}
class B : A {
    public void Metod(string s){
        Console.WriteLine(s);
    }
}
```

1. Програм има грешку, јер је метод `Metod(int i)` надјачан (предефинисан) са различитим потписом у класи B.
2. Програм има грешку, јер се `b.Metod(5)` не може позвати пошто је метод `Metod(int i)` заклоњен у класи B.
3. Програм има грешку због `b.i`, јер је поље `i` неприступачно из класе B.
4. Програм нема грешке, јер наслеђени метод класе A, `Metod(int i)` није надјачан у класи B, већ је дефинисан преоптерећен метод `Metod(string s)`.

У следећим задацима заокружите бројеве испред тражених одговора

17. Заокружити бројеве испред тражених одговора.

	/	1.5
--	---	-----

Међу понуђеним алатима, обележити **Case** алате:

1. Rational Rose
2. Oracle Designer
3. .NET
4. Microsoft Visio
5. Java
6. SQL Express

18. Заокружити бројеве испред тражених одговора.

	/	2
--	---	---

Означити операторе који се **НЕ МОГУ** користити за поређење са подупитом који враћа више вредности:

1. IN
2. BETWEEN
3. LIKE
4. SOME
5. ANY
6. ALL

19. Заокружити бројеве под којима су наведене клаузуле SQL наредбе у којима се могу користити аритметичке операције:

	/	2
--	---	---

1. SELECT
2. FROM
3. WHERE
4. ORDER BY

20. Креиране су табеле **SKOLA** и **OSNOVNASKOLA**, а затим су у табелу **SKOLA** уписани подаци извршавањем следећих наредби:

	/	2
--	---	---

```
create table Skola(  
    skolaID int primary key, Naziv nvarchar(50), gradID int, tip nvarchar(50) )  
create table OsnovnaSkola(  
    osID int primary key, Naziv nvarchar(50), gradID int )  
insert into Skola values (101, 'Nikola Tesla', 20, 'srednja strucna')  
insert into Skola values (102, 'Dusko Radovic', 20, 'osnovna')  
insert into Skola values (103, 'Sveti Sava', 30, 'osnovna')  
insert into Skola values (104, 'Bora Stankovic', 20, 'gimnazija')
```

У табелу **OSNOVNASKOLA** треба уписати податке о основним школама преписивањем потребних вредности из табеле **SKOLA**. Заокружити бројеве испред упита који ће јавити грешку при извршењу:

1. `select * into OsnovnaSkola
from Skola where tip='osnovna'`
2. `insert into OsnovnaSkola
select * from Skola where tip='osnovna'`
3. `insert into OsnovnaSkola (osID, Naziv)
select s.skolaID, s.Naziv from Skola as s where tip='osnovna'`
4. `insert into OsnovnaSkola
select s.skolaID, s.Naziv, s.gradID from Skola as s where tip='osnovna'`

21. Креиране су и попуњене подацима табеле **Korisnik** и **Prijatelji**. Њихова структура и садржај приказани су на слици:

	/	2
--	---	---

Korisnik

ID	IME	POL
1	Ana	NULL
2	Steva	m
3	Marta	z
4	Petra	z

Prijatelji

Korisnik1	Korisnik2
1	2
1	3
2	3

Извршавањем упита добија се табела са подацима.

```
select k.ime, COUNT(*) as [broj prijatelja]
from Korisnik as k
left join Prijatelji as p on p.korisnik1=k.id or p.korisnik2=k.id
where k.pol='z'
group by k.id, k.ime
```

Заокруживањем бројева испред понуђених одговора, обележити који од наведених података ће бити приказани у појединим редовима резултујуће табеле:

- | | |
|-------------|-------------|
| 1. Ana, 1 | 5. Petra, 1 |
| 2. Ana, 2 | 6. Petra, 0 |
| 3. Marta, 1 | 7. Steva, 1 |
| 4. Marta, 2 | 8. Steva, 2 |

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

	/	3
--	---	---

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

23. Наведени су искази који се односе на дефиницију **do-while** циклуса у програмском језику Ц. Заокружити бројеве испред **ТАЧНИХ** исказа:

	/	1
--	---	---

1. Прво се извршава тело циклуса, а затим израчунава вредност логичког израза. Ако се добије логичка неистина, циклус се поновно извршава.
2. Циклус се извршава барем једном.
3. Користи се када се не зна колико ће се пута циклус понављати.
4. Циклус се завршава када услов добија вредност логичке истине.

24. У програмском језику Ц је дата наредба декларације `int a, b;`

	/	1.5
--	---	-----

Имајући у виду дату декларацију, заокружити бројеве испред **НЕИСПРАВНО** написаних наредби форматираног уноса података:

1. `scanf("%d%f", &a, &b);`
2. `scanf("%d*d", &a, &b);`
3. `scanf("%d%d", &a, &b);`
4. `scanf("%d%d", a, b);`
5. `scanf("%d*d", &a);`
6. `scanf("%5d%5d", &a, &b);`

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 **ПРИКАЗ** i-тог елемента низа B:

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

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

	/	3
--	---	---

```
void Umetni(char *a, char k);
```

У main функцији дате су следеће декларације променљивих:

```
char s1[20], *s2, s3;
```

Заокружити бројеве испред исправно написаних позива декларисане функције:

1. `Umetni(s2, s1[i]);`
2. `Umetni(s2, s1);`
3. `Umetni(s2, 'A');`
4. `Umetni(s1, s3);`
5. `Umetni(*s2, s3);`
6. `Umetni(s3, &s1);`

27. У програмском језику Ц је декларисана низовна променљива:

	/	2
--	---	---

```
int niz[10];
```

Заокружити бројеве испред исправно написаних наредби читања низа целих бројева дужине 10 из бинарног фајла на који показује показивач ***in**:

1. `fread(niz, sizeof (int), 10, in);`
2. `fread(&niz, sizeof (int), 10, in);`
3. `fread(&niz, sizeof niz, 1, in);`
4. `fread(niz, sizeof niz, 1, in);`
5. `fread(niz, sizeof (niz), 1, *in);`
6. `fread(niz, sizeof (int)*10, in);`

28. Дата је дефиниција класе у програмском језику C# и састоји се од два конструктора, методе и поља `x` и `y`. У петом реду дефинисан је конструктор са параметрима који формира тачку са координатама `x` и `y`. Заокружити наредбе којима се може допунити дефиниција конструктора:

	/	2
--	---	---

```
1. public class Point {  
2.     private double x, y;  
3.     public Point() { x = 0; y = 0; }  
4.     public void set(double xx, double yy) { x = xx; y = yy; }  
5.     public Point(double x, double y) { _____; }  
6. }
```

Заокружити бројеве испред тачних одговора:

1. `set(this.x, this.y);`
2. `set(x, y);`
3. `x=this.x; y=this.y;`
4. `this.x=x; this.y=y;`
5. `x=x; y=y;`

29. Дате су наредбе у програмском језику C# које дефинишу заглавље методе `Print()` са променљивим бројем параметара. Одредити који од понуђених одговора су исправни.

	/	2
--	---	---

Заокружити бројеве испред очекиваних одговора:

1. `public void Print(params string[] niska, params double[] broj)`
2. `public void Print(params double[] broj, string niska)`
3. `public void params Print(double d1, double d2)`
4. `public void Print(params double[] broj)`
5. `public void Print(int n, params double[] broj)`

30. Дат је код на C#-у којим су креиране три класе у ланцу наслеђивања. Имајући у виду класификаторе приступа пољима класа, заокружити бројеве испред поља која ће бити видљива унутар класе Sin:

	/	1.5
--	---	-----

```
public class Deda {  
    private double penzija;  
    protected string adresa;  
    public string ime;  
}  
public class Otac: Deda {  
    private double plata;  
    protected string struka;  
}  
public class Sin: Deda {  
    public int razred;  
}
```

1. penzija
2. adresa
3. ime
4. plata
5. struka
6. razred

31. Међу понуђеним исказима заокружити оне које се односе на динамичке веб странице:

	/	3
--	---	---

1. Странице се састоје искључиво од HTML кода
2. Могу приступити базама података
3. Могу слати персонализован садржај појединачним корисницима
4. Динамичке странице се пишу у CSS језику
5. Ажурирање података на сајту је компликованије и спорије него код статичког сајта
6. Странице се извршавају на веб серверу, а резултат овог извршавања представља HTML код

Допуните следеће реченице и табеле

32. Допунити реченицу наводећи назив нормалне форме:

	/	1
--	---	---

Уколико сви некључни (споредни) атрибути релације не зависе од неког другог некључног атрибута, тј. ако не постоји транзитивна зависност било ког споредног атрибута од било ког кључа те релације, кажемо да је релација у _____ нормалној форми.

33. Дат је код написан у програмском језику Ц. Одредити вредности које ће променљиве x и y имати по извршењу следећег кода:

	/	2
--	---	---

```
int x=10;  
int y=20;  
if (x>50)  
    x-=10;  
    y+=10;
```

Уписати добијене вредности на предвиђене линије:

x = _____ y = _____

34. У програмском језуку Ц декларисане су две целобројне променљиве:

	/	2
--	---	---

```
int x=0, izbor;
```

За дате вредности променљиве **izbor**, одреди вредност променљиве **x** по извршењу следеће наредбе вишестуког гранања и уписати их на предвиђене линије:

```
switch (izbor)
{
    case 1: x += 1;
    case 2: x += 2; break;
    case 3: x += 3;
    default: x = 100;
    case 4: x += 4;
    case 5: x += 5;
}
```

1. за izbor=3, x=_____ 3. за izbor=4, x=_____
2. за izbor=10, x=_____ 4. за izbor=2, x=_____

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

	/	2
--	---	---

```
int mat[10][20]; int x, 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 **последње колоне** матрице:

```
for(x = 0; x < _____; x++)
    mat[_____][_____] *= 2;
```

36. Дата је дефиниција функције у програмском језуку Ц:

	/	3
--	---	---

```
void Transformisi(float *x, float *y, float z)
{
    z++;
    *x=*x+z;
    (*y)++;
}
```

У главном програму су декларисане променљиве и извршен је позив функције:

```
float a=10, b=10, c=10;
Transformisi(&a, &b, c);
```

Одредити које вредности имају променљиве **a**, **b** и **c** по изласку из функције и уписати их на одговарајућу линију:

a = _____

b = _____

c = _____

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

	/	3
--	---	---

```
public class KlasaA {  
    public virtual int Metod() { return 10; } }  
public class KlasaB : KlasaA {  
    public override int Metod() { return 20; } }  
public class KlasaC : KlasaB {  
    public new int Metod() { return 30; } }
```

Креирани су објекти ових класа и из њих позвана метода **Metod()**. На предвиђене линије уписати шта метод **Metod()** враћа при позиву из наведених објеката:

KlasaA a = new KlasaA(); a.Metod() враћа вредност _____

KlasaB b = new KlasaB(); b.Metod() враћа вредност _____

KlasaA bb = new KlasaB(); bb.Metod() враћа вредност _____

KlasaC c = new KlasaC(); c.Metod() враћа вредност _____

KlasaB cc = new KlasaC(); cc.Metod() враћа вредност _____

KlasaA ccc = new KlasaC(); ccc.Metod() враћа вредност _____

У следећим задацима повежите и уредите појмове према захтеву

38. Исписати на цртама испред релација редни број под којим је наведена одговарајућа кардиналност везе:

	/	2.5
--	---	-----

1. 1 : 1 _____ ВЛАСНИК – БРОЈ ТЕЛЕФОНА
2. 1 : M _____ НАСТАВНИК – ПРЕДМЕТ
3. M : M _____ ОСОБА – ПАСОШ
_____ КУПАЦ – МОДЕЛ АУТОМОБИЛА
_____ УТАКМИЦА – ГРАД ДОМАЋИН

39. Написати на цртама испред логичких операција редне бројеве њихових приоритета:

	/	1.5
--	---	-----

1. највиши приоритет _____ AND
2. средњи приоритет _____ OR
3. најнижи приоритет _____ NOT

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

	/	3
--	---	---

IDBR	IME	PREZIME	PLATA	BROD	BROD	IMEOD	MESTO
5900	Sloboda n	Golubović	900	10	50	Skladišt a	Zemun
5932	Mitar	Gavrilović	600		30	Marketin g	Vračar
5953	Persida	Kosanović	1100	20	10	Plasman	Surčin
6234	Marko	Pavlović	1300	30	20	Direkcija	Grocka
6789	Janko	Nikolić	800	10	40	Nabavka	Barajev o

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

1. **SELECT** odeljenje.imeod,
radnik.prezime
FROM odeljenje **INNER JOIN** radnik
ON radnik.brod = odeljenje.brod

Приказује све раднике (и који јесу и који нису распоређени у одељења) и само она одељења у којима има радника

2. **SELECT** odeljenje.imeod,
radnik.prezime
FROM odeljenje **LEFT JOIN** radnik
ON radnik.brod = odeljenje.brod

Приказује само одељења у којима има радника и само раднике распоређене у одељењима

3. **SELECT** odeljenje.imeod,
radnik.prezime
FROM odeljenje **RIGHT JOIN** radnik
ON radnik.brod = odeljenje.brod

Приказује сва одељења (и она у којима има и она у којима нема радника) и само оне раднике који су распоређени у одељења

41. Дата је табела **GEOGRAFIJA** која поред осталих података садржи називе градова и држава (*Naziv varchar(50)*). У зависности од услова у **WHERE** клаузули, **SELECT** упитом се приказују географски појмови из табеле. Са леве стране су дати услови нумерисани бројевима од 1 до 5, а са десне групе градова.

	/	2.5
--	---	-----

Свакој групи градова придружити по један услов уносом редног броја којим је услов нумерисан на линију испред листе градова:

- | | | |
|--------------------------------------|-------|----------------------------|
| 1. where Naziv like 'L__ %' | _____ | SIJERA LEONE, SVETA LUCIJA |
| 2. where Naziv like '___ %N%' | _____ | LA VALETA, LA KORUNJA |
| 3. where Naziv like '% L%' | _____ | EL RENO, LA KORUNJA |
| 4. where Naziv like ' _L%' | _____ | EL SALVADOR, EL RENO |
| 5. where Naziv like '___ %A' | _____ | LAS VEGAS, LOS ANGELES |

42. Са леве стране су набројани неки од прелазних знакова тј. escape секвенце, а са десне стране дати су њихови описи. На линију испред описа упишите број под којим је наведена одговарајућа escape секвенца:

	/	3
--	---	---

1. `'\a'` _____ системски звучник (bell)
2. `'\b'` _____ прелаз у нови ред (new line)
3. `'\h'` _____ враћање на почетак реда (carrage return)
4. `'\r'` _____ враћање једну курсорску позицију назад (backspace)
5. `'\n'` _____ хоризонтални табулатор (horizontal tab)
6. `'\t'` _____ није escape секвенца

43. Са леве стране дате су врсте конверзије, а са десне типови података који се користе у функцији за приказ података `printf` у програмском језику Ц. На линију испред типа података унеси број којим је означена одговарајућа конверзија:

	/	2.5
--	---	-----

1. `%d` _____ short
2. `%i` _____ signed int (u dekadnom obliku)
3. `%s` _____ long
4. `%ld` _____ unsigned
5. `%f` _____ signed int (dekadni, heksadekadni ili oktalni oblik)
6. `%e` _____
7. `%hd` _____
8. `%u` _____

44. У програмском језику Ц, декларисан је показивач на цео број и функцијом **`calloc`** додељен му је простор за смештај низа од **`n`** целих бројева:

	/	2
--	---	---

```
int *a, n;
scanf("%d", &n);
a=(int*)calloc(n, sizeof(int));
```

У левој колони дати су изрази, а у десној опис њиховог значења. На линију испред сваког од израза унеси број којим је означено одговарајуће објашњење:

- | | |
|-------------------------------|---|
| _____ <code>&a[0];</code> | 1. вредност елемента на почетној позицији у низу |
| _____ <code>*(a+n-1);</code> | 2. вредност елемента на предзадњој позицији низа |
| _____ <code>*a;</code> | 3. вредност елемента на последњој позицији у низу |
| _____ <code>a+4;</code> | 4. адреса почетног елемента низа |
| | 5. адреса четвртог елемента у низу |
| | 6. адреса петог елемента у низу |

45. На програмском језику C декларисане су променљиве:

	/	4
--	---	---

```
char s1[]="Iwnt2CmyM8sagain", *sn;
```

Са леве стране написани су изрази доделе вредности стрингу **sn**, а са десне стране понуђене су вредности стринга **sn**. На линију написати редни број под којим је наведена вредност стринга **sn** која се добија извршењем одговарајућег израза:

- | | |
|------------------------------|-----------------|
| _____ sn=strrchr(s1, 'a')-1; | 1. NULL |
| _____ sn=strchr(s1, 'a')+1; | 2. "in" |
| _____ sn=strstr(s1, "my"); | 3. "ain" |
| _____ sn=strstr(s1, "T2"); | 4. "gain" |
| | 5. "sagain" |
| | 6. "myM8sagain" |

46. Са леве стране су наведени делови/елементи класе, а са десне стране улоге појединих класних елемената. На линију испред описа улоге унети редни број под којим је наведен одговарајући елеменат класе:

	/	2.5
--	---	-----

- | | | |
|------------------------|-------|-------------------------------------|
| 1. конструктор | _____ | Опис особина објеката класе |
| 2. деструктор | _____ | Контрола приступа пољима класе |
| 3. поље (атрибут) | _____ | Опис функционалности објеката класе |
| 4. својство / property | _____ | Уништавање објеката класе |
| 5. метод | _____ | Креирање објеката класе |

47. Са десне стране су наведене су области видљивости појединих елемената класе, а са леве стране класификатори приступа којима се врши контрола области видљивости. На линију испред описа области видљивости унети редни број под којим је наведен одговарајући класификатор приступа:

	/	2
--	---	---

- | | | |
|--------------|-------|--|
| 1. private | _____ | видљив унутар класе у којој је дефинисан, као и унутар изведених класа |
| 2. public | _____ | видљив само унутар класе у којој је дефинисан |
| 3. protected | _____ | видљив унутар пројекта у коме је дефинисан |
| 4. internal | _____ | видљив и ван своје класе у којој је дефинисан |

48. Повезати појмове према захтеву.

	/	2
--	---	---

HTML документ може да прими податке од корисника помоћу форми (формулара).
Повежите дате тагове и атрибуте са њиховим дефиницијама:

- | | | |
|-----------|-------|---|
| 1. FORM | _____ | Дефинише поље за унос податка унутар HTML форме |
| 2. INPUT | _____ | Основни таг формулара са којим се креира формулар за унос података од стране корисника |
| 3. ACTION | _____ | Одређује начин на који се подаци са форме шаљу на дефинисано одредиште (може бити „post” или „get”) |
| 4. METHOD | _____ | Дефинише где проследити податке са форме после предаје (submit) форме |
-

49. У JavaScript језику многи објекти имају уграђене функције (методе) које симулирају догађаје настале услед акција корисника. Уносом редног броја методе (лева колона) на предвиђену линију, повежите метод и одговарајућу акцију корисника.

	/	2
--	---	---

- | | | |
|-----------|-------|----------------------------------|
| 1. submit | _____ | Учита страницу у прегледач |
| 2. focus | _____ | Изврши слање форме |
| 3. blur | _____ | Уђе у фокус неког елемента форме |
| 4. load | _____ | Изађе из фокуса елемента форме |
-

50. Направите редослед потребних корака за приказ статичке веб странице уносом редних бројева на линије испред описа корака, почев од броја 1 до броја 5.

	/	3
--	---	---

- | | |
|-------|---|
| _____ | Веб сервер шаље пронађену страну клијенту - веб претраживачу. |
| _____ | Корисник захтева да види веб страну (кликом на линк, укуцавањем адресе у адресну линију веб претраживача и слично) |
| _____ | Аутор је креирао страну која се састоји од HTML кода и ставио је на веб сервер. |
| _____ | Веб претраживач обрађује добијени HTML код и приказује кориснику уредно форматирану страницу са свим елементима (сликама, линковима, табелама, ...) |
| _____ | Веб сервер проналази HTML страну коју је корисник захтевао |
-
-