

## ЧАСТ 1 (Време за работа: 90 минути)

Отговорите на задачите от 1. до 16. включително отбелязвайте в листа за отговори!

### 1. Кои от принципите на ООП са реализирани в показания код?

```
public class Rectangle
{
    public double GetArea(double length, double width)
    {
        return length * width;
    }
}
public class Square : Rectangle
{
    public double GetArea(double size)
    {
        return size * size;
    }
}
```

- A) капсулиране и наследяване  
B) полиморфизъм и абстракция

- Б) наследяване и полиморфизъм  
Г) наследяване и абстракция.

### 2. Какво ще изведе следния код?

```
public class Person
{
    public void Introduction(int age, string name)
    {
        Console.WriteLine("My name is " + name); ;
    }
    public void Introduction(string name, int age = 0)
    {
        Console.WriteLine($"My name is {name} and I am {age} years old.");
    }
}
static void Main(string[] args)
{
    Person p = new Person();
    p.Introduction("Steve");
}
```

- A) My name is Steve  
B) съобщение за грешка

- Б) My name is Steve and I am 0 years old.  
Г) нито едно от посочените

### 3. Каква е разликата между интерфейс и абстрактен клас?

- A) интерфейсът може да съдържа само сигнатура на методи, докато абстрактният клас може да съдържа и имплементация на методи  
Б) интерфейсът съдържа методи с тяхната имплементация, докато абстрактният клас може да съдържа методи само със сигнатура и методи с имплементация  
B) членовете на интерфейса могат да имат различни модификатори за достъп за разлика от абстрактните класове

Г) нито едно от изброените

**4. Кое число ще се добави последно към списъчното поле след изпълнение на програмния фрагмент?**

```
for(int i = 23; i <= 38; i = i + 3)
{
    listBox1.Items.Add(i);
}
```

- A) 37                      Б) 38                      В) 35                      Г) Това е безкраен цикъл.

**5. Какво ще изведе програмният фрагмент след изпълнението си?**

```
int a = 2, b = 3, c = 4;

a = ++b * c--;
c /= b--;
b += a++;

Console.WriteLine(a + " " + b + " " + c);
```

- A) 17 3 4                      Б) 16 1 2                      В) 16 19 2                      Г) 17 19 0

**6. Кое от изброените е коректният начин да намерим индекса на втория символ 's' в стринга "She sells sea shells on the sea-shore"?**

A)	<pre>string str = "She sells sea shells on the sea-shore"; int i; i = str.SecondIndexOf("s"); Console.WriteLine(i);</pre>
Б)	<pre>string str = "She sells sea shells on the sea-shore"; int i, j; i = str.FirstIndexOf("s"); j = str.IndexOf("s", i + 1); Console.WriteLine(j);</pre>
В)	<pre>string str = "She sells sea shells on the sea-shore"; int i, j; i = str.IndexOf("s"); j = str.IndexOf("s", i + 1); Console.WriteLine(j);</pre>
Г)	<pre>string str = "She sells sea shells on the sea-shore"; int i, j; i = str.LastIndexOf("s"); j = str.IndexOf("s", i - 1); Console.WriteLine(j);</pre>

**7. Как се проверява дали даден елемент (item) се съдържа в стек или опашка?**

- A) Contains(item)                      Б) Peek(item)                      В) IsContains(item)                      Г) Find(item)

**8. Кои са ключовите думи, свързани с прихващане на изключения в C#?**

- A) try, catch, valid, invalid                      Б) try, valid, finally, throw  
 В) try, catch, finally, throw                      Г) finally, throw, valid, invalid

**9. В какво се изразява операцията селекция в SQL заявка?**

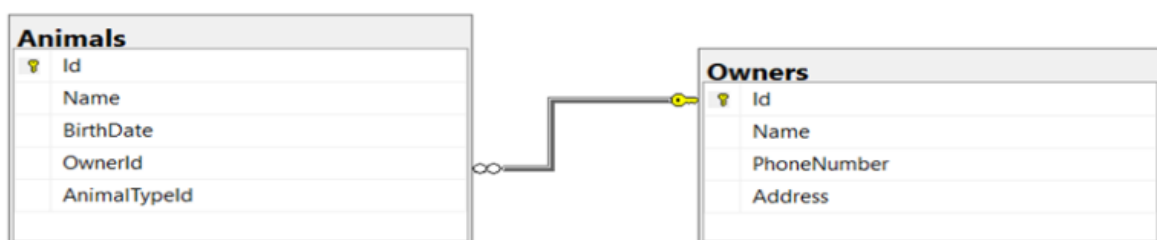
- А) избиране на определени записи от таблицата  
 Б) съединяване на данни от различни таблици  
 В) групиране на данните според определен критерий  
 Г) сортиране на данните във възходящ ред

**10. Какво ще изведе програмният фрагмент след изпълнението си?**

```
int a, b;
int c = 10;
int d = 12;
int e = 5;
int f = 6;
a = c * (d + e) / f + d;
b = c * (d + e / f + d);
if(a<b)
{
    Console.WriteLine("parentheses changes value");
}
else if(a>b)
{
    Console.WriteLine("they have same value");
}
```

- А) parentheses changes values  
 Б) they have same value  
 В) двете променливи имат равни стойности и няма да се изведе нищо  
 Г) нито едно от изброените

**11. Дадена е следната диаграма на релационна база от данни:**



Кое от твърденията не е вярно?

- А) **Id**, **Name** и **Address** са колони в таблица **Owners**  
 Б) **OwnerId** е външен ключ в таблицата **Owners**  
 В) всяка стойност в полето **Id** на таблицата **Owners** е уникална  
 Г) **Id** и **OwnerId** са първични ключове в таблицата **Animals**

## 12. Какво ще изведе програмният фрагмент?

```
var arlist = new ArrayList();
arlist.Add(1);
arlist.Add("Bill");
arlist.Add(" ");
arlist.Add(true);
arlist.Add(4.5);
Console.WriteLine(arlist[4]);
```

A) true

Б) Bill

В) 4.5

Г) съобщение за грешка, защото не може в arlist да се съхраняват данни от различен тип

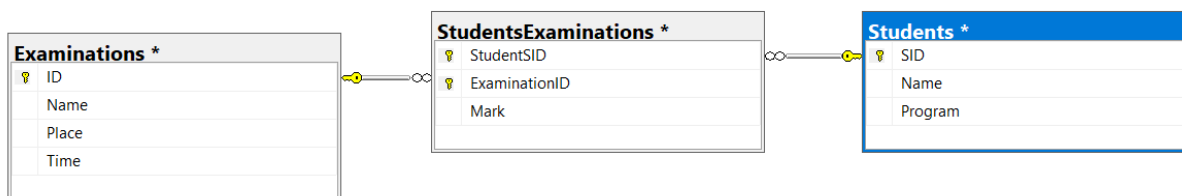
## 13. Дадена е заявката

```
SELECT ID, FirstName, LastName, Salary
FROM Employees
ORDER BY Salary
```

Коя от следващите таблици е резултат от горната заявка?

A)	<table border="1"> <thead> <tr> <th>ID</th><th>FirstName</th><th>LastName</th><th>Salary</th></tr> </thead> <tbody> <tr><td>1</td><td>Иван</td><td>Иванов</td><td>8000.00</td></tr> <tr><td>2</td><td>Георги</td><td>Георгиев</td><td>3500.00</td></tr> <tr><td>3</td><td>Стефан</td><td>Попов</td><td>1500.00</td></tr> <tr><td>4</td><td>Димитър</td><td>Димитров</td><td>2500.00</td></tr> <tr><td>5</td><td>Даниел</td><td>Ангелов</td><td>1800.00</td></tr> </tbody> </table>	ID	FirstName	LastName	Salary	1	Иван	Иванов	8000.00	2	Георги	Георгиев	3500.00	3	Стефан	Попов	1500.00	4	Димитър	Димитров	2500.00	5	Даниел	Ангелов	1800.00	Б)	<table border="1"> <thead> <tr> <th>ID</th><th>FirstName</th><th>LastName</th><th>Salary</th></tr> </thead> <tbody> <tr><td>3</td><td>Стефан</td><td>Попов</td><td>1500.00</td></tr> <tr><td>5</td><td>Даниел</td><td>Ангелов</td><td>1800.00</td></tr> <tr><td>4</td><td>Димитър</td><td>Димитров</td><td>2500.00</td></tr> <tr><td>2</td><td>Георги</td><td>Георгиев</td><td>3500.00</td></tr> <tr><td>1</td><td>Иван</td><td>Иванов</td><td>8000.00</td></tr> </tbody> </table>	ID	FirstName	LastName	Salary	3	Стефан	Попов	1500.00	5	Даниел	Ангелов	1800.00	4	Димитър	Димитров	2500.00	2	Георги	Георгиев	3500.00	1	Иван	Иванов	8000.00
ID	FirstName	LastName	Salary																																																
1	Иван	Иванов	8000.00																																																
2	Георги	Георгиев	3500.00																																																
3	Стефан	Попов	1500.00																																																
4	Димитър	Димитров	2500.00																																																
5	Даниел	Ангелов	1800.00																																																
ID	FirstName	LastName	Salary																																																
3	Стефан	Попов	1500.00																																																
5	Даниел	Ангелов	1800.00																																																
4	Димитър	Димитров	2500.00																																																
2	Георги	Георгиев	3500.00																																																
1	Иван	Иванов	8000.00																																																
В)	<table border="1"> <thead> <tr> <th>ID</th><th>FirstName</th><th>LastName</th><th>Salary</th></tr> </thead> <tbody> <tr><td>1</td><td>Иван</td><td>Иванов</td><td>8000.00</td></tr> <tr><td>2</td><td>Георги</td><td>Георгиев</td><td>3500.00</td></tr> <tr><td>4</td><td>Димитър</td><td>Димитров</td><td>2500.00</td></tr> <tr><td>5</td><td>Даниел</td><td>Ангелов</td><td>1800.00</td></tr> <tr><td>3</td><td>Стефан</td><td>Попов</td><td>1500.00</td></tr> </tbody> </table>	ID	FirstName	LastName	Salary	1	Иван	Иванов	8000.00	2	Георги	Георгиев	3500.00	4	Димитър	Димитров	2500.00	5	Даниел	Ангелов	1800.00	3	Стефан	Попов	1500.00	Г)	<table border="1"> <thead> <tr> <th>ID</th><th>FirstName</th><th>LastName</th><th>Salary</th></tr> </thead> <tbody> <tr><td>3</td><td>Стефан</td><td>Попов</td><td>1500.00</td></tr> <tr><td>1</td><td>Иван</td><td>Иванов</td><td>8000.00</td></tr> <tr><td>4</td><td>Димитър</td><td>Димитров</td><td>2500.00</td></tr> <tr><td>5</td><td>Даниел</td><td>Ангелов</td><td>1800.00</td></tr> <tr><td>2</td><td>Георги</td><td>Георгиев</td><td>3500.00</td></tr> </tbody> </table>	ID	FirstName	LastName	Salary	3	Стефан	Попов	1500.00	1	Иван	Иванов	8000.00	4	Димитър	Димитров	2500.00	5	Даниел	Ангелов	1800.00	2	Георги	Георгиев	3500.00
ID	FirstName	LastName	Salary																																																
1	Иван	Иванов	8000.00																																																
2	Георги	Георгиев	3500.00																																																
4	Димитър	Димитров	2500.00																																																
5	Даниел	Ангелов	1800.00																																																
3	Стефан	Попов	1500.00																																																
ID	FirstName	LastName	Salary																																																
3	Стефан	Попов	1500.00																																																
1	Иван	Иванов	8000.00																																																
4	Димитър	Димитров	2500.00																																																
5	Даниел	Ангелов	1800.00																																																
2	Георги	Георгиев	3500.00																																																

## 14. Релационна база данни е представена със следната диаграма:



В таблиците е въведена информация.

### Students

SID	Name	Program
1	Иван Иванов Иванов	Информатика
2	Иван Иванов Стоянов	Информатика
3	Иван Стоянов Иванов	Информатика
4	Иван Иванв Иванв	Математика
5	Петър Иванов Иванов	Математика

### Examinations

ID	Name	Place	Time
1	Информатика	ППМГ	2023-06-12 08:45:00.000
2	Математика	ППМГ	2023-06-13 09:00:00.000
3	Химия	ГПЧЕ	2023-06-14 08:00:00.000
4	Информатика	ГПЧЕ	2023-06-12 08:10:00.000

### StudentsExaminations

	StudentSID	ExaminationID	Mark
	1	1	4.20
	1	2	5.00
	1	3	3.75
	2	1	3.00
	2	2	4.13
	2	3	6.00
	3	1	5.00
	3	3	6.00
	3	4	5.25
	4	1	3.50
	4	2	5.87
	4	4	5.00
	5	1	4.12
	5	3	4.00
	5	4	6.00

### Резултатна таблица

	Name	Grade
1	Математика	5.000000
2	Химия	4.937500
3	Информатика	4.508750

С коя заявка ще получите списък по предмети и средната оценка, постигната на изпита по всеки предмет, като списъкът е нареден в намаляващ ред по тази оценка?

A)	SELECT Name, AVG(StudentsExaminations.Mark) AS Grade FROM Examinations GROUP BY Examinations.ID ORDER BY AVG(StudentsExaminations.Mark) DESC	B)	SELECT Name, AVG(Mark) AS Grade FROM Examinations LEFT JOIN StudentsExaminations ON StudentsExaminations.ExaminationID = Examinations.ID ORDER BY AVG(Mark) DESC
Б)	SELECT Name, AVG(Mark) AS Grade FROM Examinations INNER JOIN StudentsExaminations ON StudentsExaminations.ExaminationID = Examinations.ID GROUP BY Examinations.Name ORDER BY AVG(Mark) DESC	Г)	SELECT Name, AVG(Mark) AS Grade FROM Examinations RIGHT JOIN StudentsExaminations ON StudentsExaminations.ExaminationID = Examinations.ID ORDER BY AVG(Mark) DESC

15. Дадена е релацията Students. Степента и мощността на релацията са:

SID	Name	Program
1	Иван Иванов Иванов	Информатика
2	Иван Иванов Стоянов	Информатика
3	Иван Стоянов Иванов	Информатика
4	Иван Иванв Иванв	Математика
5	Петър Иванов Иванов	Математика

A) 2 и 5      Б) 3 и 6      В) 3 и 5      Г) 3 и 4

16. Кое от изброените не е модел на база от данни?

A) йерархичен      Б) релационен      В) файлов      Г) мрежов

Отговорите на задачите от 17. до 24. вкл. запишете в листа за отговори!

**17. Каква ще бъде стойността на променливата x след изпълнението на следния програмен фрагмент?**

```
static void Main(string[] args)
{
    int x = 1;
    int y = 2;
    int z = 3;

    x = F(y, z);
    Console.WriteLine(x);
}

static int F(int a, int b)
{
    if(a==0)
    {
        return b + 1;
    }
    else if(b==0)
    {
        return F(a - 1, a);
    }
    else
    {
        return F(a - 1, F(a, b - 1));
    }
}
```

**18. Запишете в листа за отговори какво ще се изведе на стандартния изход, при изпълнението на следния програмен фрагмент?**

```
List<int> numbers = new List<int>() { 10, 20, 30, 40, 35, 25, 15 };
int n = numbers.Count;
for (int i = 0; i < n / 2; i++)
{
    int p = numbers[i];
    numbers[i] = numbers[n - 1 - i];
    numbers[n - 1 - i] = p;
}
Console.Write("[");
Console.Write(string.Join(", ", numbers));
Console.WriteLine("]");
```

**19. В опашка последователно са постъпили елементи със стойност 14, 2, -3, 0, -2, 1, 9, а в стек: 4 и -4. От опашката се изважда елемент и ако той е положителен, се добавя в стека. След това стека премахва един елемент. Действието се повтаря до изваждането на елемент със стойност 0 от опашката.**

**В листа за отговори срещу числото 1 запишете кои ще бъдат елементите на стека, а срещу 2 – кои ще са елементите на опашката след приключване на действията.**

**20. Даден е програмен фрагмент, който трябва да намира сбора от цифрите на дадено число n. В листа за отговори запишете пропуснатия код.**

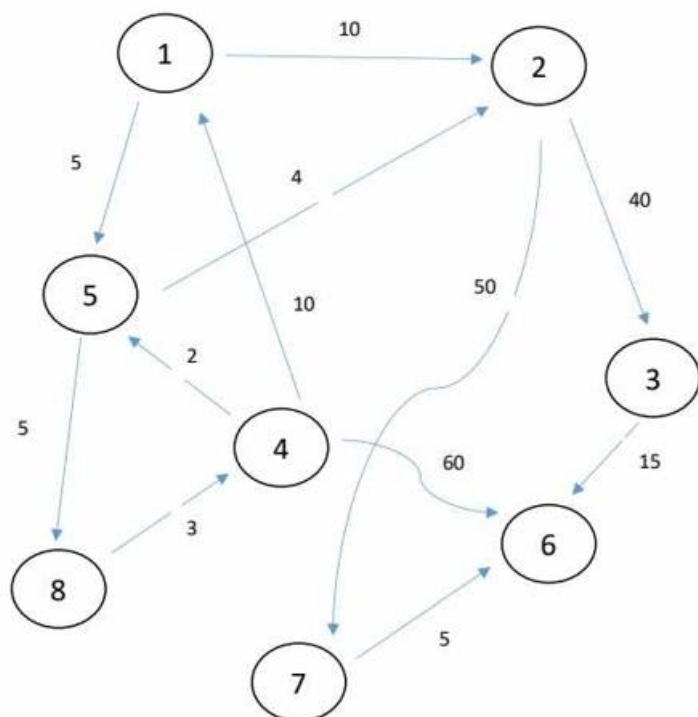
```
int n = int.Parse(Console.ReadLine());
int sum = 0;

while (n > 0)
{
    ...
}
Console.WriteLine(sum);
```

**21. Определете резултата от изпълнението на следната програма при въвеждане на числата 12 и 13.**

```
string result = "";
int[] numbers = { 4, 5, 6, 1, 8, 56, 55, 3, 2, 21 };
Console.WriteLine("Въведете две цели числа:");
try
{
    int a = int.Parse(Console.ReadLine());
    int b = int.Parse(Console.ReadLine());
    int s = numbers[a] + numbers[b];
    Console.WriteLine(s);
}
catch (FormatException)
{
    result = "Вие не въведохте число";
    Console.WriteLine(result);
}
catch (IndexOutOfRangeException)
{
    result = "Излязохте извън границите на масива";
    Console.WriteLine(result);
}
```

## 22. Даден е граф



Да се опише обхождане в ширина на дадения граф с начален връх 4.

23. Даден е програмен фрагмент, който намира сбора на елементите над главния диагонал и броя на нечетните елементи под второстепенния диагонал в квадратна таблица от числа. В кода са пропуснати части, които са означени с (1), (2) и (3). В листа за отговори срещу числата 1, 2 и 3 запишете пропуснатия код, а срещу 4 запишете какво ще изведе последната команда в дадения код.

```
int[,] a = new int[4, 4] {  
    {0, 1, 2, 3},  
    {4, 5, 6, 7},  
    {8, 9, 10, 11},  
    {12, 13, 14, 15}  
};  
int sum = 0; int br = 0;  
for (int i = 0; i < 4; i++)  
{  
    for (int j = 0; j < 4; j++)  
    {  
        if (1) sum += a[i, j];  
        if (2) && (3) br++;  
    }  
}  
Console.WriteLine(sum + " " + br);
```



**24. Даден е програмен фрагмент, който завършва с четири оператора за извеждане на конзолата. В листа за отговори срещу всяко от числата 1, 2, 3 и 4 запишете какво ще изведе съответната команда.**

```
Dictionary<int, string> d = new Dictionary<int, string>();  
d.Add(1, "red");  
d.Add(2, "pink");  
d.Add(3, "purple");  
d.Add(7, "black");  
Console.WriteLine(d.TryGetValue(3, out string s));  
Console.WriteLine(d.TryGetValue(6, out string s1));  
Console.WriteLine(s);  
Console.WriteLine(s1);
```

## ЧАСТ 2 (Време за работа: 150 минути)

Файловете с отговорите на задачите от 25. до 28. включително, запишете в изпитната система като спазите указанията в условието на задачата!

Внимание! Имената на работните файлове, които прикачвате в изпитната система НЕ трябва да съдържат текстове или символи, които могат да доведат до нарушаване на анонимността на изпитната Ви работа!

25. Създайте проект с име `Zad_25`, в който се създава списък от числа. От клавиатурата се въвеждат в произволен ред командите **Add**, **Insert**, **Remove**, **Shift left**, **Shift right** до получаване на командата **End**. Реализирайте всяка команда с метод.

- Методът **Add** получава като параметри списък с числа и число (**number**), което трябва да се добави в края на списъка. Връща като резултат промененият списък.
- Методът **Insert** получава като параметри списък, число за вмъкване (**number**) и позиция на която трябва да се вмъкне това число (**index**) и връща като резултат списък с вмъкнатото на зададената позиция число.
- Методът **Remove** получава като параметри списък и число (**index**) – позицията на елемент, който трябва да се премахне и връща като резултат промененият списък.
- **Shift left** – този метод получава като параметър списък и число (**count**). Първият елемент на списъка трябва да се премести в края му **count** пъти. Методът връща като резултат промененият списък.
- **Shift right** – този метод получава като параметър списък и число (**count**). Последният елемент става първи **count** пъти. Методът връща като резултат промененият списък.

Възможно е полученият като параметър индекс да бъде извън границите на колекцията. Тогава трябва да изведете съобщение **"Invalid index"**.

Примерен вход	Примерен изход
1 23 29 18 43 21 20 Add 5 Remove 5 Shift left 3 Shift left 1 End	43 20 5 1 23 29 18
5 12 42 95 32 1 Insert 3 0 Remove 10 Insert 8 6 Shift right 1 Shift left 2 End	Invalid index 5 12 42 95 32 8 1 3

28. Създайте клас **Car** (автомобил), представящ автомобил със следните характеристики и поведение:

Характеристики

- Марка: свойство **Brand**

- Година на производство: свойство **Year**
- Пробег (изминати километри): свойство **Mileage**
- Цена по ценова листа на производител: свойство **Value**

Поведение

- Метод **Price** за изчисляване на актуална цена по следните критерии:
  - Автомобили до 3 години: 80% от цената по ценова листа на производителя;
  - Автомобили между 4 и 6 години: 60% от цената по ценова листа на производителя;
  - Автомобили на 7 и повече години: 30% от цената по ценова листа на производителя.

Създайте клас **Truck** (камион), наследяващ от класа автомобил и дефиниращ следните допълнителни характеристики и поведение:

Характеристики

- Вместимост (тонаж): свойство **Tonnage**

Поведение

- Метод **Price** за изчисляване на актуална цена по следните критерии:
  - Камион до 5 години - 100% от цената по ценова листа на производителя;
  - Камион над 5 години с вместимост до 5 тона - 30% от цената по ценова листа на производителя;
  - Камион над 5 години с вместимост между 5 и 10 тона - 60% от цената по ценова листа на производителя;
  - Камион над 5 години с вместимост над 10 тона - 80% от цената по ценова листа на производителя.

Напишете програма, която въвежда от текстов файл data.txt, който се намира в папката на изпълнимия файл, данните за автомобили и камиони. Всяко превозно средство се описва на три реда:

- Ред 1: цифрата 1, ако следва описание на лек автомобил, или 2, ако следва описание на камион;
- Ред 2: марка – непразна редица от латински букви и цифри, евентуално съдържаща интервали;
- Ред 3: три или четири положителни числа, разделени с точно един интервал. Първото от тях е годината на производство на превозното средство; второто – пробега му в цели километри; третото – реално число с до два знака след десетичната точка: цената на производителя в евро. Четвърто число на този ред има само ако се описва камион и отразява неговия тонаж в цели тонове.

Програмата трябва да създава в оперативната памет линейна структура от описаните във входа превозни средства и да извежда на стандартния изход списък на въведените данни в ред, обратен на въвеждането. Всеки ред от списъка описва едно превозно средство и има следния формат:

марка, двоеточие, интервал, пробег, интервал, km, запетая, интервал, актуална цена, закръглена и форматирана до втория знак след десетичния разделител.

#### ПРИМЕР (data.txt)

```
1
Tesla
2019 10000 110000
2
Volvo
2005 360000 250000 14
```

#### ИЗХОД

```
Volvo: 360000 km, 200000,00
Tesla: 10000 km, 88000,00
```