1. Коя от следните стойности може да се присвои на променлива от тип **ushort**?
   1. -567
   2. **0.8**
   3. 67500
   4. '5'
2. Какъв резултат ще се изведе на стандартния изход след изпълнение на следния програмен фрагмент:

int a = 3, b = 2, c;

a += b; = 5

c = --a+7 ; = 11

c %= 6; = 0

double d = c / b + Math.Round(7.49 / b, 2); = 3.74

Console.WriteLine(a + " " + c + " " + d);

1. **5 0 3.74**
2. 4 5 5.74
3. 4 5 5.75
4. 4 0 3.75
5. Какъв тип данни връща методът Next() на класа **System.Random** при генериране на случайно число?
6. **int**
7. double
8. float
9. string
10. Какъв резултат ще се изведе на стандартния изход след изпълнение на следния програмен фрагмент:

int number = 127;

int count = 0;

do

{

if(number<20)

{

number /=3;

}

else

{

number-=20; = 107

}

count++; = 1

}

while (number > 1);

Console.WriteLine($"{count} -> {number}");

1. 7 -> 2
2. 7 -> 0
3. 8 -> 0
4. 6 -> 1
5. Ако знаете, че ASCII кодът на буквата 'a' е 97, кой от низовте ще се изведе на стандартния изход след изпълнение на следния програмен фрагмент:

static void Main(string[] args)

{

Console.WriteLine(Encode("amazing"));

}

static string Encode(string text)

{

string encodedText = "";

foreach (char c in text)

{

int asciiValue = (int)c;

int encodedAsciiValue = asciiValue + 1;

if (char.IsLower(c) && encodedAsciiValue > 122)

{

encodedAsciiValue -= 26;

}

encodedText += (char)encodedAsciiValue; | bnbajoh

}

return encodedText;

}

1. **bnbajoh**
2. cocbkpi
3. bnbajog
4. cocblpi
5. Какъв резултат ще се изведе на стандартния изход след изпълнение на следния програмен фрагмент:

List<int> nums = new List<int>() { 25, 14, 32, 39, 18, 23 };

HashSet<int> set = new HashSet<int>();

for (int i = 0; i < nums.Count; i++)

{

nums[i]%=5;

set.Add(nums[i]);

}

Console.WriteLine(set.Sum()); | 0, 4, 2, 4, 3, 3

1. **16**
2. 31
3. 12
4. 9
5. Какъв резултат ще се изведе на стандартния изход след изпълнение на следния програмен фрагмент:

private static void Main(string[] args)

{

Console.WriteLine(CountWays(9));

}

static int CountWays(int amount)

{

if (amount == 0) return 1;

if (amount < 0) return 0;

return CountWays(amount - 5) + CountWays(amount - 2);

}

1. 1
2. 2
3. 3
4. 4
5. Каква е причината за грешката в следния програмен фрагмент?

List<int> list = new List<int>();

for (char i = 'A'; i < 'J'; i++)

{

list.Add(i);

}

foreach (char sign in list)

{

Console.WriteLine(sign);

list.Remove(sign);

}

1. **Не може към списък с елементи от тип int да добавяме елементи от тип char;**
2. Не може с цикъл **foreach** да се обхождат елементи на списък от тип **char**;
3. Не може да се променя списакът по време на итерация на **foreach** цикъла;
4. Не може да изтриваме елементи от списък;
5. Трябва да създадете приложение с графичен потребителски интерфейс, в което човек трябва да попълни информация. Необходимо е да определи какви занимания са му любими, като избере колкото желае от 5 посочени възможност. С какви елементи на потребителския интерфейс да се визуализират заниманията?
   * 1. радио бутони (radio buttons)
     2. **полета за отметка (check boxes)**
     3. списъчно поле (list box)
     4. падащ списък (combo box)
6. Какъв резултат ще се изведе на стандартния изход след изпълнение на следния програмен фрагмент:

private static void Main(string[] args)

{

int[,] matrix = {

{11, 2, 3, 7},

{4, 15, 6, 12},

{7, 18, 9, 10},

{8, 19, 11, 13}

};

int count = 0;

int n= matrix.GetLength(0);

for (int i = 0; i < n; i++)

{

if (matrix[i, n-i-1]%10 %2==0)

{

count++;

}

}

Console.WriteLine(count);

}

1. 0
2. 1
3. **3**
4. 7
5. В проект е създаден клас със статичен метод:

class Sequence

{

public static int Fibonacci(int n)

{

if (n <= 1)

{

return n;

}

else

{

return Fibonacci(n - 1) + Fibonacci(n - 2);

}

}

}

Какъв код да се напише метода Main в Program.cs, така че резултатът от този метод за стойност на параметъра 7 да се изведе на стандартния изход?

а. private static void Main(string[] args)

{

Console.WriteLine(Sequence.Fibonacci(7));

}

б. private static void Main(string[] args)

{

Sequence sequence = new Sequence();

Console.WriteLine(sequence.Fibonacci(7));

}

**в. private static void Main(string[] args)**

**{**

**Console.WriteLine(Fibonacci(7));**

**}**

г. private static void Main(string[] args)

{

Console.WriteLine(Sequence(7));

}

1. На кой етап от жизнения цикъл на информационна система се избират технологиите, които ще се използват и стандарта за писане на код?
   1. анализ и дефиниция
   2. **проектиране**
   3. разработка
   4. тестване

1. Релационна база от данни за туристическа агенция съдържа няколко таблици. Едната от таблиците дава информация за клиентите на агенцията, а друга за нейните дестинации на пътувания. За да се отрази правилно дейността на агенцията, тези две таблици трябва:
2. да са свързани с връзка „едно към едно“;
3. да са свързани с връзка „едно към много“;
4. **да са свързани с връзка „много към много“ чрез допълнителна таблица;**
5. да не бъдат свързвани.
6. В таблицата rivers, съдържаща се в база от данни са въведени следните записи:

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| id | name | length | drainage\_area | outflow |
| 1 | Nile | 6650 | 3254555 | Mediterranean |
| 2 | Amazon | 6400 | 7050000 | Atlantic Ocean |
| 3 | Yangtze | 6300 | 1800000 | East China Sea |
| 4 | Mississippi | 6275 | 2980000 | Gulf of Mexico |
| 5 | Lena | 4400 | 2490000 | Laptev Sea |
| 6 | Congo | 4700 | 3680000 | Atlantic Ocean |
| 7 | Amur | 4444 | 1855000 | Sea of Okhotsk |
| 8 | Volga | 3645 | 1380000 | Caspian Sea |

Какъв ще бъде резултатът от изпълнението на заявката:

SELECT id,name FROM rivers

WHERE length NOT BETWEEN 6200 AND 6500 AND outflow Like '%Sea%'

ORDER BY drainage\_area ASC;

a.

|  |  |
| --- | --- |
| id | name |
| 8 | Volga |
| 7 | Amur |
| 5 | Lena |
| 1 | Nile |
| 6 | Congo |

**б.**

|  |  |
| --- | --- |
| **id** | **name** |
| **-8** | **Volga** |
| **10** | **Amur** |
| **5** | **Lena** |

в.

|  |  |
| --- | --- |
| id | name |
| 5 | Lena |
| 10 | Amur |
| 8 | Volga |

г.

|  |  |
| --- | --- |
| id | name |
| 6 | Congo |
| 1 | Nile |
| 5 | Lena |
| 7 | Amur |
| 8 | Volga |

1. Каква е основната разлика между ArrayList и обикновенния масив (array) в C#?
   1. ArrayList може да съдържа елементи само от типове string и int, докато обикновният масив е ограничен до елементи от един и същи тип
   2. ArrayList поддържа генеричност, докато обикновният масив не
   3. ArrayList е част от пространството от имена System.Collections, докато обикновният масив е част от пространството от имена System.Collections.Generic
   4. **ArrayList автоматично разширява размера си при добавяне на нови елементи, докато обикновният масив има фиксиран размер**
2. Коя UML диаграма отговаря на дефинирания клас?

public class Medicine

{

private string name;

private double price;

private int year;

public Medicine(string name, double price)

{

this.name = name;

this.price = price;

this.year = DateTime.Now.Year;

}

protected void DecreasePrice(double percent)

{

double decreaseAmount = price \* percent / 100;

price -= decreaseAmount;

}

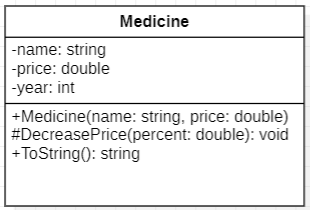
public override string ToString()

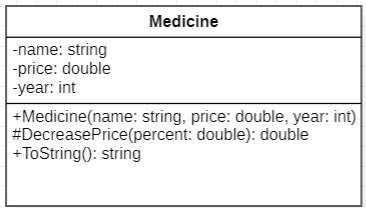
{

return $"Medicine {name} with price {price:F2}, from year {year}";

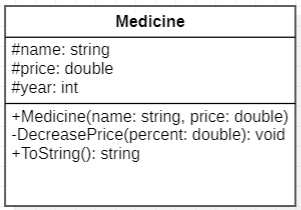
}

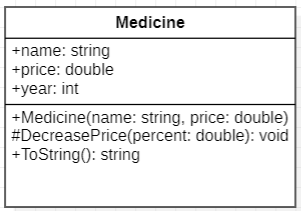
}

* + - 1. 

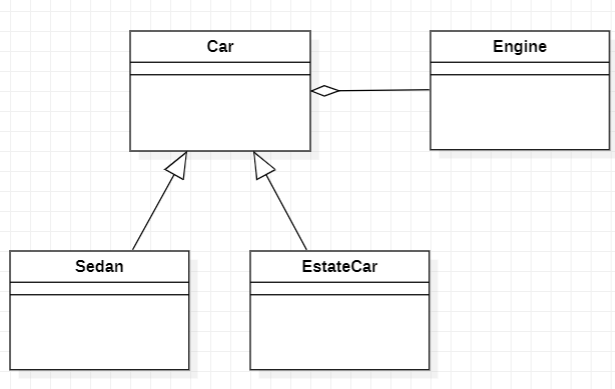


б.

 в.



г.

1. В диаграмата са онагледени връзките между класовете Car, Engine, Sedan и EstateCar. Определете дали типа на връзката е “има” (has a) или “e” (is a) между посочените класове.

(1) връзката между Car и Engine : has-a

(2) връзката между Car и EstateCar : is-a

(3) връзката между Sedan и Engine : has-a

1. При изпълнение на записания по-долу код се извежда резултат на три отделни реда. Запишете какво се извежда срещу съответната цифра, записана преди всяко извеждане:

Queue<int> queueElements = new Queue<int>();

Stack<int> stackElements = new Stack<int>();

for (int i = 1; i < 20; i+=3)

{

stackElements.Push(i);

}

/\* (1) \*/ Console.WriteLine(string.Join(" ", stackElements));

int count=stackElements.Count;

for (int i = 1; i < count; i++)

{

queueElements.Enqueue(stackElements.Pop() + stackElements.Count);

}

/\* (2) \*/Console.WriteLine(string.Join(" ", queueElements));

/\* (3) \*/Console.WriteLine(queueElements.Peek());

Запишете вашите отговори:

**(1)19 16 13 10 7 4 1**

**(2)26 22 18 14 10 6 2**

**(3)26**

1. При изпълнение на записания по-долу код се извежда резултат на три отделни реда. Запишете какво се извежда срещу съответната цифра, записана преди всяко извеждане:

private static void Main(string[] args)

{

int[] arr = { 1, 5, 6, 10, 13, 16, 20, 21, 22, 25 };

int number = 17;

bool isFind = false;

int count = 0;

int left = 0;

int right = arr.Length - 1;

while (left <= right)

{

int mid = left + (right - left) / 2;

if (arr[mid] == number)

{

isFind = true;

}

else if (arr[mid] < number)

{

left = mid + 1; count++;

}

else

{

right = mid - 1;

}

}

/\* (1) \*/ Console.WriteLine(isFind);

/\* (2) \*/ Console.WriteLine(count);

/\* (3) \*/ Console.WriteLine(right);

}

Запишете вашите отговори:

**(1) False**

**(2) 2**

**(3) 6**

1. В проект е създаден клас **TV**, който се наследява от класа **SmartTV**. В клас **TV** е дефиниран метод **CalculatePrice**, който се използва **само** в класа **SmartTV**.

class TV

{

protected double screenSizeInInches;

public TV(double screenSizeInInches)

{

this.screenSizeInInches = screenSizeInInches;

}

(**protected**) decimal CalculatePrice()

{

decimal price = (decimal)screenSizeInInches \* 20;

return price;

}

}

class SmartTV : TV

{

private string smartFeatures;

public SmartTV(string smartFeatures, double screenSizeInInches) :base(screenSizeInInches)

{

this.smartFeatures = smartFeatures;

}

(**public**) decimal CalculatePrice()

{

decimal price=base.CalculatePrice();

price += 200;

return price;

}

}

private static void Main(string[] args)

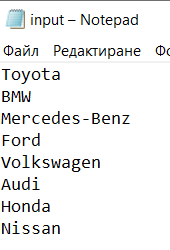
{

SmartTV tV=new SmartTV("",50);

Console.WriteLine(tV.CalculatePrice());

}

* + 1. Като спазвате принципа за капсулация и се съобразите с предоставената функционалност, определете най-подходящия модификатор за видимост за метода CalculatePrice от класа **TV**.
    2. Като спазвате принципа за капсулация и се съобразите с предоставената функционалност, определете най-подходящия модификатор за видимост за метода CalculatePrice от класа **SmartTV**.
    3. Каква стойност ще се изведе на стандартния изход? - **1200**

1. Какво ще се запише във файла output.txt, ако във файлът input.txt, който е записан в папката на изпълнимия файл ca записани следните марки автомобили:

Изпълнява се кодът от програмния фрагмент:

using (StreamReader reader = new StreamReader("input.txt"))

{

using(StreamWriter writer = new StreamWriter("output.txt"))

{

int count = 0;

string brand = reader.ReadLine();

while (brand != null)

{

if(brand.ToLower().Contains('n'))

{

writer.Write($"{brand} ");

}

else if(brand.Length > 5)

{

count++;

}

brand = reader.ReadLine();

}

writer.WriteLine();

writer.Write(count);

}

}

* + - 1. Какво се извежда на първия ред? – Mercedes Benz Volskwagen Honda Nissan
      2. Какво се извежда на втория ред? - 1

1. Напишете какво трябва да се попълни в програмния фрагмент на обозначените с цифри места. Определете какво е предназначението на метода **CountNumbers.**

static void Main(string[] args)

{

try

{

int count = CountNumbers(70, 100);

Console.WriteLine(count);

}

catch (ArgumentException ex)

{

Console.WriteLine(.......(1).......);

}

}

static int CountNumbers (int start, int end)

{

if (start >= end)

{

..............(2)………………;

}

int count = 0;

for (int i = start; i <= end; i++)

{

int number = i;

while (number != 0)

{

if (number % 10 == 7)

{

count++;

break;

}

number /= 10;

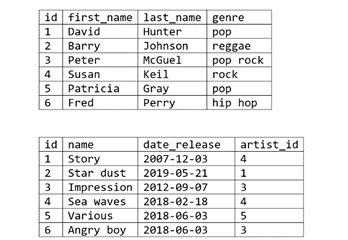
}

}

return count;

}

1. **ex.Message**
2. **throw new ArgumentException(“Invalid number range!”);**
3. **Методът CountNumbers намира числата в даден интервал, които като се разделят на 10 остатъка им да е равен на 7 и записва броя на тези числа.**
4. В база от данни са създадени две таблици, свързани с връзка едно към много, защото всеки албум е на един изпълнител. В таблиците са попълнени следните данни:



Изпълненa е заявката:

SELECT concat(ar.first\_name,' ',ar.last\_name) AS 'Artist name',al.name AS 'Album name',al.date\_release

FROM artists AS ar

JOIN albums AS al

ON ar.id=al.artist\_id

WHERE al.date\_release>'2014-1-1'

ORDER BY date\_release;

Запишете отговорите си срещу съответната цифра:

(1) колко ще бъдат редовете в резултата от изпълнението на заявката?

(2) колко ще бъдат колоните в резултата от изпълнението на заявката?

(3) какво ще се изведе в клетката от първи ред и първа колона на резултата?

(4) какво ще се изведе в клетката от трети ред и втора колона на резултата?

**(1) 4**

**(2) 3**

**(3) Susan Keil**

**(4) Various**

1. Дефинирани са класовете Plant и Flower. Попълнете пропуснатото в кода на отбелязаните с цифри места:

public class Plant

{

public string Name { get; set; }

public string Color { get; set; }

public Plant(string name, string color)

{

Name = name;

Color = color;

}

public ........(1)........ void Grow()

{

Console.WriteLine($"Растението {Name} расте.");

}

}

public class Flower : Plant

{

public string Type { get; set; }

public Flower(string name, string color, string type) : base(.........(2)..........)

{

Type = type;

}

public ........(3)........ void Grow()

{

Console.WriteLine($"Цветето {Name} расте и цъфти.");

}

}

**(1) static**

**(2) string name, string color**

**(3) static**

1. Създадена е таблица products с полета за наименование на продукт (name), категория (category) и цена (price). Трябва да се напише SQL заявка, която да групира продуктите по категория и да изчисли средната цена на продуктите във всяка категория. Попълнете липсващите части от заявката, означени с цифри:

SELECT category, .....(1)..... AS AvgPrice

FROM .....(2).....

GROUP BY .....(3).....;

**(1) AVG(Price)**

**(2) products**

**(3) category**

**Задача 26**

Напишете програма, която прочита от стандартния вход две цели числа **a** и **b, които са по-големи от 10**. Програмата да намира и извежда на стандартния изход всички числа в интервала [a, b], чиито цифри са подредени в нарастващ ред, без да са равни.

**Вход:** От стандартния вход коректно се въвеждат:

* На първия ред на стандартния вход се въвежда цяло положително число a в интервала [10, 100000];
* На втория ред на стандартния вход се въвежда цяло положително число b в интервала [10,100000]

**Изход:**

Изведете на стандартния изход на един ред:

- Ако коректно е зададен интервала [a, b]:

“Числата са: .....................................................................”

- Ако НЕ е зададен коректно интервала [a, b]

“Задайте коректен интервал, необходимо е a<b”

и се дава възможност за ново въвеждане на стойностите на a и b

**Пример:**

|  |  |
| --- | --- |
| **Вход:** | **Изход:** |
| 120  160 | Числата са: 123 124 125 126 127 128 129 134 135 136 137 138 139 145 146 147 148 149 156 157 158 159 |
| 65  30  30  65 | Задайте коректен интервал, необходимо е a<b  Числата са: 34 35 36 37 38 39 45 46 47 48 49 56 57 58 59 |

**Задача 27**

Създайте база данни за болница **pastry\_shop**, съдържаща две таблици **types\_sweets** и **sweets**.

27.1. Създайте 2 заявки, с които да създадете двете таблици:

Таблицата **types\_sweets** със следните атрибути:

id – цяло число, първичен ключ;

types\_name – текст;

description– цяло число;

Таблицата **sweets** със следните атрибути:

id – цяло число, първичен ключ;

name – текст;

availability – цяло число;price – реално число;

types\_id – цяло число, Връзка с таблицата **types\_sweets**.

27.2. Създайте заявки, които добавят в таблицата **types\_sweets** следните записи:

|  |  |  |
| --- | --- | --- |
| id | types\_name | description |
| 1 | Cake | Traditional cakes for special occasions |
| 2 | Tart | Crispy crusts filled with sweet fruit filling |
| 3 | Cupcake | Single-serving cakes topped with frosting |
| 4 | Muffin | Individual baked goods, often with fruit |
| 5 | Pastry | Flaky pastries filled with cream or fruit |

Създайте заявки, които добавят в таблицата **sweets** следните записи:

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| id | name | availability | price | types\_id |
| 1 | Lemon Tart | 8 | 18 | 2 |
| 2 | Red Velvet Cupcake | 20 | 4 | 3 |
| 3 | Strawberry Shortcake | 5 | 30 | 1 |
| 4 | Blueberry Muffin | 15 | 3.80 | 4 |
| 5 | Vanilla Cream Puff | 12 | 3.20 | 5 |
| 6 | Chocolate Cake | 10 | 25 | 1 |

27.3. Създайте заявка, която да изведе име (name) и наличност (availability) на сладкишите, които са с цена от 10 до 18 лева включително. Резултатът да бъде подреден в намаляващ ред на цената;

27.4. Създайте заявка, която да изведе име на сладкиш (name), цена (price) и име на типа сладкиш (types\_name) за всеки сладкиш;

27.5. Създайте заявка, която да изведе име на типа сладкиш (types\_name) и сума от цените на сладкишите от ВСЕКИ тип, подредени по азбучен ред на типовете.

27.6. Създайте заявка, която да изведе име на сладкиша (name), име на типа (types\_name) и общата цена, която се получава при умножаване на наличността по цената и се увеличи с 20 % за ДДС.

**Задача 28:**

Създайте конзолно приложение с име zad28.

28.1 Дефинирайте клас Exam

А) Класът има следните характеристики:

* subject – низ,
* date – дата,
* quotient – реално число,
* grade – реално число,
* type – низ, може да бъде едно от следните: state matriculation, candidate student, language, external assessment

Б) Класът да има конструктор, който задава стойности на характеристиките.

В) Дефинирайте метод **FinalGrade**, който изчислява крайна оцента от изпита по следния начин: quotient \* grade

Г) Пренапишете метода **ToString**, така че да връща низа:

*"The exam in the subject of {subject} dated {date} has a final grade of FinalGrade}."*

Дробните числа да са закръглени до два знака в дробната част.

28.2 Валидиране на данните за класа Exam:

А) Предметът (subject) не може да бъде празен низ. Ако е празен низ, се извежда: "The subject can not be empty."

Б) Коефициентът трябва да е в интервала (0,1]. Ако е извън този интервал се извежда съобщение "Specify a quotient from 0 to 1."

В) Типът на изпита може да е само една от следните: state matriculation, candidate student, language, external assessment. Ако не е, се извежда съобщение: "The type can be only state matriculation, candidate student, language or external assessment."

28.3 Създайте клас MyExams

А) Класът съдържа свойства:

* name – име на ученика
* exams - списък от изпити.

Б) Класът да има конструктор с един аргумент за името на ученика.

В) В класа дефинирайте методи за:

• Добавяне на нов изпит – Add (Exam exam) – изпитът се добавя към списъка. Не се допуска добавяне на изпит, ако вече има изпит на тази дата. При успешно добавяне се извежда съобщение "You have successfully added a exam.", а когато не може да се добави се извежда "There is an exam on that date."

• Премахване на изпит – RemoveLowestScore () – премахва от списъка изпита, който е с най-ниска крайна оценка. Ако има няколко изпита с една и съща най-ниска оценка, се премахват всички и се извежда съобщение "{count} exams have been removed".

• Промяна на датата на определен изпит ChangeDate (DateTime oldDate, DateTime newDate)

• Определяне на броя на изпитите от определен тип – CountExams (string type).

• Пренапишете метода ToString, така че да връща на нов ред низа от метода ToString за всеки изпит.

28.4 Създайте програма, в която създавате инстанция на класа MyExams с име на ученик, въведено на стандартния вход. Програмата може да приема последователно и многократно някоя от командите и да изпълнява съответните действия, като използва методите от класа MyExams:

• Add <предмет> <дата> <коефициент> <оценка> <тип> - добавяне на нов изпит в списъка;

• RemoveLowestScore – списъка изпита, който е с най-ниска крайна оценка и извежда на стандартния изход "{count} exams have been removed".

• ChangeDate <стара дата> <нова дата>- промяна на датата на определен изпит. Прието е, че на всяка дата има само един изпит, така че се променя датата само на един изпит.

• CountExams <тип> - извежда на стандартния изход броя на изпитите от посочения тип

• Print – извежда на стандартния изход данните за всички изпити, като за всеки изпит се извеждат на нов ред.

Всички дробни числа да са закръглени до два знака в дробната част.

След всяка команда се извежда резултатът от нея. Задаването на команди приключва с въвеждане на команда "Close".