

ДЪРЖАВЕН ЗРЕЛОСТЕН ИЗПИТ ПО
ИНФОРМАТИКА
ПРОФИЛИРАНА ПОДГОТОВКА
23 август 2024 г.
ВАРИАНТ 2

ЧАСТ 1 (Време за работа: 90 минути)

Внимание! Полетата за отговори НЕ трябва да съдържат текстове или символи, които могат да доведат до нарушаване на анонимността на изпитната Ви работа!

За задачите от 1. до 16. включително изберете точно един от отговорите!

1. Композиция (composition) в обектно ориентираното програмиране означава, че:

- А) полетата на класа трябва да бъдат декларирани като частни (private)
- Б) един клас наследява друг клас
- В) променлива от родителския клас се отнася до обект от наследяващия клас
- Г) даден клас съдържа поле, което е обект от друг клас

2. Коя от представените UML диаграми илюстрира класа, дефиниран с дадения код?

C#
<pre>class Car { private int engineVolume; private int yearProd; private string model; private string brand; public void go() {} public void stop() {} public void changeDirection(int d){} }</pre>
JAVA
<pre>class Car { private int engineVolume; private int yearProd; private String model; private String brand;</pre>

```
public void go() {}
public void stop() {}
public void changeDirection(int d){}
}
```

A)

Car
-engineVolume : double -model : string -brand : string -yearProd : int
+go() +stop() +changeDirection(d: int)

Б)

Car
-engineVolume : int -model : string -brand : string -yearProd : int
+go() : int +stop() +changeDirection(d: int)

B)

Car
-engineVolume : int -model : string -brand : string -yearProd : int
+go() +stop() +changeDirection

Г)

Car
-engineVolume : int -model : string -brand : string -yearProd : int

```
+ go()
+ stop()
+ changeDirection( d: int)
```

3. Колко пъти ще се изпълни следващия цикъл и кое е последното изведено число?

C#	JAVA
<pre>int i = 5; while (i <= 5) { Console.WriteLine(i); i += 2; }</pre>	<pre>int i = 5; while (i <= 5) { System.out.println(i); i += 2; }</pre>

- A) Цикълът е безкраен
- Б) 1 път, последното изведено число е 7
- В) 0 пъти, не се извежда нищо
- Г) 1 път, последното изведено число е 5

4. Какъв ще бъде резултатът след изпълнение на програмния код?

C#
<pre>public class Student { static int studentG; private string name; private int group; public Student(string name, int group) { this.name = name; this.group = group; studentG++; } public static void Main() { Student studentA = new Student("George", 3); Student studentB = new Student("Miranda", 4); Student studentC = new Student("Alex", 1); Student studentD = new Student("Dalia", 2); Console.WriteLine(Student.studentG); } }</pre>
Java
<pre>public class Student { static int studentG; private String name ; private int group;</pre>

```
public Student(String name, int group) {
    this.name = name;
    this.group = group;

    studentG++;
}

public static void main(String[] args) {
    Student studentA = new Student("George", 3);
    Student studentB = new Student("Miranda", 4);
    Student studentC = new Student("Alex", 1);
    Student studentD = new Student("Dalia", 2);

    System.out.println(Student.studentG);
}
}
```

- A) 1 Б) 2 В) 3 Г) 4

5. На кой ред запазените думи са само модификатори за достъп?

- A) public, class, final
- Б) public, static, protected
- В) private, protected, public
- Г) private, overload, public

6. Даденият програмен код трябва да извежда средноаритметичното на стойностите на масива. На кой ред от програмния фрагмент е допусната грешка?

C#
1 int[] numbers = { 5, -10, 17, -20, 25 }; 2 int sum = 0; 3 foreach (int number in numbers) { 4 sum += number; 5 } 6 double average = sum / numbers.Length; 7 Console.WriteLine("Average: {0:F2}", average);
Java
1 int[] numbers = { 5, -10, 17, -20, 25 }; 2 int sum = 0; 3 for (int number : numbers) { 4 sum += number; 5 } 6 double average = sum / numbers.length; 7 System.out.println(String.format("Average: %.2f", average));

- A) На ред 6
- Б) На ред 7
- В) Няма грешка в кода
- Г) На ред 3

7. Коя от фигурите ще бъде изведена на стандартния изход след извикването на метода Fig(3) (за C#) или fig(3) (за Java)?

C#
<pre> static void Row (int number) { if (number > 0) { Console.Write(number + " "); Row(number - 1); } } static void Fig(int number) { if (number > 0) { Fig(number - 1); Row(number); Console.WriteLine(); } } </pre>
Java
<pre> public static void row (int number){ if (number > 0){ System.out.print(number + " "); row (number - 1); } } public static void fig (int number){ if (number > 0){ fig (number - 1); row (number); System.out.println(); } } </pre>

- A) 1 2 3
- 12
- 1

- Б) 3 2 1
2 1
1
- В) 1
1 2
1 2 3
- Г) 1
2 1
3 2 1

8. Какво ще се изведе на стандартния изход, след изпълнение на програмния фрагмент?

C#
<pre>string str = " C# Programming Course"; string sub = str.Substring(4, 11); int index1 = str.IndexOf("C#"); int index2 = str.IndexOf("r", 7); Console.WriteLine(sub + " " + index1 + " " + index2);</pre>
Java
<pre>String str = " C# Programming Course"; String sub = str.substring(4, 15); int index1 = str.indexOf("C#"); int index2 = str.indexOf("r", 7); System.out.println(sub + " " + index1 + " " + index2);</pre>

- A) Programming 0 6
- Б) Programming 1 8
- В) Programming 0 7
- Г) C# Programming 1 7

9. Кой алгоритъм е представен в програмния фрагмент, ако се знае, че масивът array е подреден във възходящ ред?

C#
<pre>static int Function (int[] array, int left, int right, int num) { if (left > right) return -1; int middle = (left + right) / 2; if (num == array[middle]) return middle; if (num < array[middle]) return Function (array, left, middle - 1, num); else return Function (array, middle + 1, right, num); }</pre>
Java
<pre>static int function(int[] array, int left, int right, int num) {</pre>

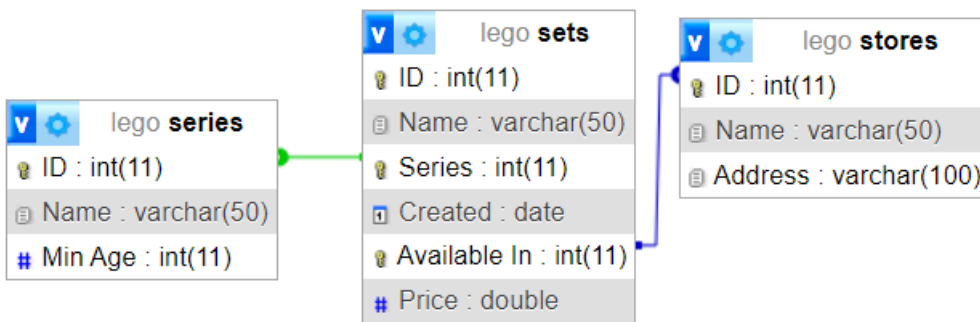
```
if (left > right) return -1;
int middle = (left + right) / 2;
if (num == array[middle]) return middle;
if (num < array[middle])
    return function(array, left, middle - 1, num);
else
    return function(array, middle + 1, right, num);
}
```

- A) Сливане на сортирани редици
- Б) Бързо сортиране в низходящ ред
- В) Двоично търсене
- Г) Комбинации

10. През кой етап от жизнения цикъл на една информационна система се разработва времеви график и план за изпълнение на проекта

- A) Проектиране
- Б) Разработка
- В) Експлоатация
- Г) Анализ

11. Кое е вярното твърдение за дадената диаграма?



- A) Връзката между таблиците lego stores - lego sets е 1 : n (едно към много).
- Б) Връзката между таблиците lego stores - lego sets е n : n (много към много).
- В) Атрибутът Series на таблицата lego sets е външен ключ за таблицата lego series.
- Г) Атрибутът ID на таблицата lego stores е външен ключ за таблицата lego sets.

12. Даден е текстовия файл Numbers.txt, съдържащ произволен брой цели числа, всяко на нов ред. Какъв ще бъде резултатът от изпълнението на следния програмен фрагмент?

```
C#
static void Main(string[] args)
{
```

```

int[] array = { 1, 2, 15 };
foreach (int el in array)
{
    Console.WriteLine(el);
}
StreamReader reader = new StreamReader("Numbers");
string line = reader.ReadLine();
while (line != null)
{
    int num = Convert.ToInt32(line);
    Console.WriteLine(num);
    line = reader.ReadLine();
}
}

```

Java

```

public static void main(String[] args) throws FileNotFoundException {
    int[] array = {1, 2, 15};
    for (int el : array) {
        System.out.println(el);
    }
    Scanner scanner = new Scanner(new File("Numbers"));
    while (scanner.hasNext()) {
        int num = scanner.nextInt();
        System.out.println(num);
    }
}

```

- A) Програмата няма да стартира, ще даде грешка – грешно име на файл.
- B) Програмата ще стартира, но ще прекъсне със съобщение за грешка, че файлът не може да бъде намерен.
- B) Програмата ще стартира и ще отвори файл Numbers.txt за четене.
- Г) Програмата ще стартира, ще създаде файл Numbers и ще го отвори за четене.

13. В база от данни в таблицата meals (дадена по-долу) са въведени следните записи:

Id	Name	Category	Quantity	Price
1	chicken	Protein	200	12,50
2	rice	Grain	150	4,50
3	broccoli	Vegetable	200	3,80
4	pasta	Grain	230	7,50
5	potato	Grain	360	5,50
6	chocolate	Dessert	100	4,50
7	pork	Protein	250	15,60
8	fish	Protein	200	10,10
9	peas	Vegetable	150	2,50
10	apple	Dessert	80	1,50
11	tofu	Protein	200	9,70

12	milk	Beverage	200	3,00
----	------	----------	-----	------

Какъв ще бъде резултатът от изпълнението на дадената заявка?

```
SELECT Category, COUNT(Id) AS Count_of_meals FROM meals
GROUP BY Category
HAVING COUNT(Id)>=2
ORDER BY Count_of_meals, Category DESC
```

A)

Category	Count_of_meals
Vegetable	2
Dessert	2
Grain	3
Protein	4

Б)

Category	Count_of_meals
Protein	4
Grain	3
Vegetable	2
Dessert	2

В)

Category	Count_of_meals
Grain	3
Protein	4

Г)

Category	Count_of_meals
Protein	4
Grain	3

14. Посочете правилната последователност на описаните абстрактни структури от данни.

Линейната абстрактна структура от данни е структура от данни, която се използва за съхранение на последователност от еднотипни елементи, като елементите се добавят и премахват на място в структурата, където е необходимо.

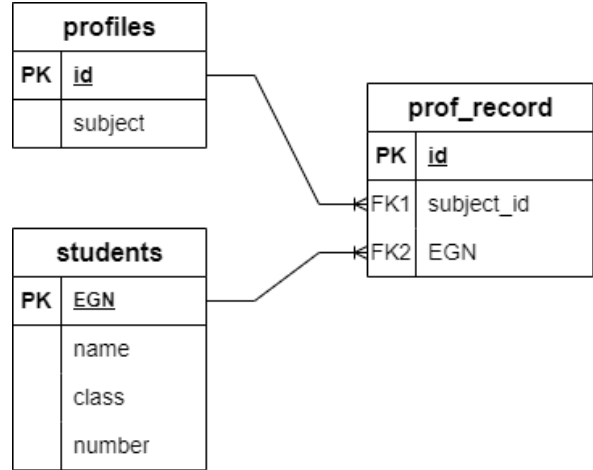
Линейната абстрактна структура от данни е структура от данни, която се използва за съхранение на последователност от еднотипни елементи, като елементите се добавят и премахват в съответствие с принципа "първи влязъл, първи излязъл" (First In, First Out).

Линейната абстрактна структура от данни е структура от данни, която се използва за съхранение на последователност от еднотипни елементи, като елементите се добавят и премахват в съответствие с принципа „последен влязъл, първи излязъл“ (Last In, First Out).

- A) стек, опашка, списък
- Б) опашка, списък, стек
- В) списък, опашка, стек
- Г) опашка, стек, списък

15. Дадена е следната диаграма на реляционна база от данни:

Кои от твърденията са верни за връзките между таблиците в базата данни?



1. Връзката между таблица students и таблица prof_record е едно към много.
 2. Връзката между таблица students и таблица prof_record е едно към едно.
 3. Връзката между таблица profiles и таблица prof_record е едно към много
 4. Връзката между таблица profiles и таблица prof_record е много към много.
 5. Няма връзка между таблица students и таблица profiles.
 6. Връзката между таблица students и таблица profiles е много към много.
- A) 1, 4, 5
 - Б) 1, 3, 6
 - В) 2, 3, 5
 - Г) 2, 4, 6

16. Добавянето на кои два реда, на празното място в дадения програмен фрагмент, ще доведе до извеждане на стандартния изход думата „nation”, като резултат от изпълнението на фрагмента?

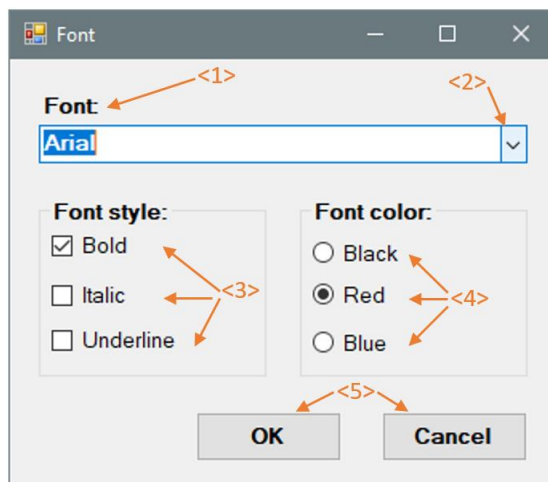
C#
<pre>var str = new StringBuilder(); Console.WriteLine(str);</pre>
Java
<pre>var str = new StringBuilder(); System.out.println(str);</pre>

C#	Java
A) str.Add("examination"); str.Remove(0, 5);	A) str.add("examination"); str.delete(0, 5);
Б) str.Add("examination");	Б) str.add("examination");

<pre>str.Remove(0, 4);</pre>	<pre>str.delete(0, 4);</pre>
<p>В) str.Append("examination"); str.Remove(0, 5);</p>	<p>В) str.append("examination"); str.delete(0, 5);</p>
<p>Г) str.Append("examination"); str.Remove(0, 4);</p>	<p>Г) str.append("examination"); str.delete(0, 4);</p>

Отговорите на задачите от 17. до 24. включително запишете в полетата за отговори под задачата!

17. В полетата за отговори срещу всеки номер напишете визуален компонент, отговарящ на номера върху даденото изображение, който може да бъде използван за изграждането на представения на изображението потребителски интерфейс.



18. Довършете изреченията, като в полетата за отговори срещу А) и Б) запишете типа връзка is-a (е), has-a (има) или няма връзка.

А) Когато класа А наследява класа В, казваме, че класовете А и В са свързани с връзка от тип

Б) Когато поле на класа А е препратка (референция) към клас В, казваме, че класовете А и В са свързани с връзка от тип

19. В празна опашка последователно са постъпили елементите 1, 20, 1, 5, 0, 10, а в празен стек – 0 и 6. От опашката се изваждат два елемента, първият от които (ако е различен от 0) се добавя отново в опашката, а вторият от тях се добавя в стека. Действието се повтаря до изваждането на елемент със стойност 0 от опашката (няма значение дали е първи или втори от двойката).

В полетата за отговори срещу (1), (2), (3) и (4) запишете:

- (1) Колко са елементите в стека?
- (2) Колко е броят на елементите в опашката?
- (3) Кои са елементите на стека?
- (4) Кои са елементите на опашката?

Забележка: Винаги се изваждат два елемента от опашката, дори ако единият от тях е 0.

20. Даденият програмен фрагмент трябва да извежда на стандартния изход всички прости делители (без повторения) на естествено число, въведено от стандартния вход, но има изпуснати части, отбелязани с номера <1>, <2>, <3>, <4>, <5> и <6>.

Срещу номерата <1>, <2>, <3>, <4>, <5> и <6> запишете пропуснатите части от кода, така че да се получи верен програмен код.

C#
<pre>int number = ...<1>...; for (int i = 2; ...<2>...; ...<3>...) { if (...<4>...) Console.Write(i + " "); while (...<5>...) { ...<6>...; } }</pre>
Java
<pre>Scanner scr = new Scanner(System.in); int number = ...<1>...; for (int i = 2; ...<2>...; ...<3>...) { if(...<4>...) System.out.print(i + " "); while (...<5>...) { ...<6>...; } }</pre>

21. В полето за отговори запишете какво ще се изведе на стандартния изход след изпълнението на програмния фрагмент?

C#
<pre>public class Test { public static void Main(string[] args) { try { Console.WriteLine("Welcome to Java"); int i = 0; int y = 2 / i; Console.WriteLine("Welcome to C#"); } } }</pre>

```
        catch (DivideByZeroException ex)
        {
            Console.WriteLine("Welcome to Java and C#");
        }
        finally
        {
            Console.WriteLine("End of the block");
        }
    }
}

JAVA
public class Test {
    public static void main(String[] args) {
        try {
            System.out.println("Welcome to Java");
            int i = 0;
            int y = 2 / i;
            System.out.println("Welcome to C#");
        }
        catch (RuntimeException ex) {
            System.out.println("Welcome to Java and C#");
        }
        finally {
            System.out.println("End of the block");
        }
    }
}
```

22. Програмният фрагмент трябва да въвежда от стандартния вход размера на квадратен масив от цели числа с не повече от 20 реда и 20 колони, както и неговите елементи. От стандартния вход коректно се въвеждат на първи ред – броят n на редовете/колониите, а следващите n реда съдържат по n цели числа, не по-големи от 1000000 по абсолютна стойност, разделени с интервал. Фрагментът трябва да намира и извежда на стандартния изход сумата от нечетните стойности на елементите на вторичния диагонал на двумерния масив. В кода има допуснати грешки. В полето за отговор запишете на кои редове има грешка и срещу номера на реда запишете верния код.

Забележка: Вторичен диагонал – диагоналът на матрицата, започващ от горен десен елемент и стигащ до долен ляв елемент.

C#
<pre> 1 int n = Convert.ToInt32(Console.ReadLine()); 2 int sum = 0; 3 int[,] arr = new int[20, 20]; 4 for (int i = 0; i < n; i++) { 5 string[] line = Console.ReadLine().Split(); 6 for (int i = 0; i < n; i++) { 7 arr[i, j] = Convert.ToInt32(line[j]); 8 } 9 } 10 for (int i = 0; i < n; i++) { 11 if (arr[i, i] % 2 == 1) { 12 sum += arr[i, i]; 13 } 14 } 15 Console.WriteLine(sum); </pre>
Java
<pre> 1 Scanner scr = new Scanner(System.in); 2 int n = scr.nextInt(); 3 int sum = 0; 4 int[][] arr = new int[20][20]; 5 for (int i = 0; i < n; i++) { 6 for (int i = 0; i < n; i++) { 7 arr[i][j] = scr.nextInt(); 8 } 9 } 10 for (int i = 0; i < n; i++) { 11 if(arr[i][i] % 2 == 1) { 12 sum += arr[i][i]; 13 } 14 } 15 System.out.println(sum); </pre>

23. В полето за отговор запишете по какво трябва да се различават методите на даден клас, за да ги наричаме претоварени (overload) методи?

- (1) Брой на аргументите
- (2) Имената на методите
- (3) Подредбата (позицията) на аргументите, които са от различен тип
- (4) Тип на аргументите
- (5) Тип на връщания от метода резултат

24. За всеки клас от представените в полетата за отговори, обозначени с <1>, <2> и <3> запишете метода, който добавя елемент след последния към създадената инстанция.

C#
<pre>List<String> list = new List<String>(); list. ...<1>... ("Maria"); Stack<String> stack = new Stack<String>(); stack. ...<2>... ("Maria"); Queue<String> queue = new Queue<String>(); queue. ...<3>... ("Maria");</pre>
Java
<pre>List<String> list = new ArrayList<>(); list. ...<1>... ("Maria"); Stack<String> stack = new Stack<>(); stack. ...<2>... ("Maria"); Queue<String> queue = new ArrayDeque<>(); queue. ...<3>... ("Maria");</pre>

25. На изображението са дадени таблиците Students и Profiles от релационна база с данни и резултат (таблицата Query result) от дадената по-долу заявка:

Students

Id	FirstName	LastName	Points	ProfileId
1	Иван	Иванов	200	1
2	Петър	Петров	150	2
3	Стефан	Стефанов	280	3
4	Симеон	Симеонов	120	4
5	Гергана	Георгиева	250	5
6	Таня	Тодорова	180	1
7	Петя	Попова	300	2
8	Ангел	Ангелов	220	3
9	Ани	Атанасова	160	4
10	Георги	Георгиев	200	5

Profiles

Id	Name
1	Математика
2	Математика и информатика
3	Информатика
4	Информатика и ИТ
5	Графичен дизайн

Query result

ProfileId	Max_Points
1	200
2	300
3	280
4	160
5	250

```
SELECT ProfileId, MAX(Points) AS Max_Points
FROM Students
GROUP BY ProfileId
```

В полето за отговори напишете модификация на дадената заявка, така че като резултат да се добави още една колона, в която се визуализира името на профила.

ДЪРЖАВЕН ЗРЕЛОСТЕН ИЗПИТ ПО
ИНФОРМАТИКА
ПРОФИЛИРАНА ПОДГОТОВКА
23 август 2024 г.
ВАРИАНТ 2

ЧАСТ 2 (Време за работа: 150 минути)

Файловете с отговорите на задачите от 26. до 28. включително, качете 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!

26. Управител на IT компания получава доклад за печалбата ѝ за n ($n \leq 100$) месеца назад в хиляди лева. Той иска да разбере най-големия брой последователни месеци, през които компанията е имала печалба не по-ниска от тази на предходния месец и с колко процента най-ниската печалба за периода на доклада е по-малка от печалбата за следващия месец.

Създайте приложение с име *Zad26*, което чете от стандартния вход два реда: от първия ред се чете едно естествено число n – броят на месеците в доклада и от втория ред - n естествени числа, разделени с точно един интервал – печалбите (в хиляди лева). Приложението намира най-големия брой последователни месеци, през които компанията е имала печалба не по-малка от тази на предходния месец и с колко процента най-ниската печалба за периода на доклада е по-малка от печалбата за следващия месец. Ако най-ниската печалба се достига в последния за доклада месец, то тя се сравнява с предпоследния месец от доклада. Гарантирано е, че има само един месец с най-ниска печалба и входните данни се подават коректно.

Приложението извежда на стандартния изход търсената информация в следния формат:

На първия ред се извежда: The longest period with bigger profit is <брой последователни месеци> months.

На втория ред се извежда: Smaller with <процент> %.

Дробните числа да се формират до втория знак след десетичната запетая.

Примерен вход:

10

5 3 4 6 7 1 2 3 4 5

Изход:

Longest period with bigger profit is 5 mounths.

Smaller with 50.00%

**Забележки:* Приемат се и решения с графичен потребителски интерфейс (ГПИ), в които данните се въвеждат в подходящи контроли, а резултатът се извежда в етикет или нередактируемо текстово поле.

Прикачете в изпитната система архив с име `inf_23.08.2024_zad26.zip`, съдържащ файловете с Вашите решения.

27. Създайте база от данни с име **Travel**, която съхранява данни за екскурзии до различни дестинации.

А) Напишете заявка, чрез която в базата данни се създава таблицата **Countries** със следните атрибути:

- **ID** – цяло число, автоматично увеличаващо се с 1, първичен ключ
- **Name** – текст до 30 знака.

Б) Напишете заявка, чрез която в базата данни се създава таблицата **Destinations** със следните атрибути:

- **ID** – цяло число, автоматично увеличаващо се с 1, първичен ключ
- **Town** – текст до 30 знака.
- **Distance** – цяло число;
- **Duration** – цяло число;
- **Price** – реално число до 10 знака с 2 знака след десетичната запетаята;
- **CountryId** – цяло число, външен ключ към таблица **Countries**.

В) Напишете заявки за добавяне на следните кортежи в таблица **Countries** (данните за таблицата са в ресурсен файл с име `resources.txt`):

ID	Name
1	France

2	Germany
3	Italy
4	Spain
5	Austria

Г) Напишете заявки за добавяне на следните кортежи в таблица **Destinations** (данните за таблицата са в ресурсен файл с име resources.txt):

ID	Town	Distance	Duration	Price	CountryId
1	Paris	2169	4	1800.00	1
2	Berlin	4006	6	2100.00	2
3	Rome	1666	3	1500.00	3
4	Madrid	2966	7	1800.00	4
5	Milan	1408	4	1900.00	3
6	Venice	1152	3	1200.00	3
7	Barcelona	2375	7	2800.00	4

Д) Напишете заявка, която променя цената на пътуването до Paris на 1700.

Е) Напишете заявка, която изтрива от таблицата Destinations всички записи, чийто имена на градове започват с 'Ba'

Ж) Напишете заявка, която връща името на града и разстоянието, за тези дестинации, които са на разстояние между 1500 и 2500 (включително).

З) Напишете заявка, която връща името на града и цената на най-скъпата дестинация.

И) Напишете заявка, която намира за всяка страна броя на дестинациите в нея, подредени в намаляващ ред по брой пътувания. При равен брой на пътувания, данните да се подредят по име на държавата в нарастващ ред.

Прикачете в изпитната система файл с име inf_23.08.2024_zad27.zip, съдържащ създадената база от данни (при работа с MS Access) или заявките за създаването на база от данни (при работа със стандартен език за заявки) и написаните от Вас търсени заявки.

28. Във футболен клуб работят директори, треньори и футболисти.

Създайте приложение с име **Zad28**, което имплементира базов клас ClubMember и негови наследници FootballPlayer (футболист), Director (директор) и Coach (треньор). Класовете трябва да съдържат следните характеристики и свойства:

А) Класът ClubMember с полета:

- firstName (име) - низ
- lastName (фамилия) - низ
- age (възраст) – цяло число
- salary (заплата) – реално число

Класът трябва да притежава следните методи:

- **конструктор**, който задава стойности на всички характеристики;
- **абстрактен метод** Info (за C#) и info (за Java), който извежда на стандартния изход информацията за съответния служител;

Данните за класа ClubMember да се валидират, съгласно следните ограничения:

- Имената на членовете на клуба не могат да са празни низове иначе да се генерира изключение със съобщението: „The name can't be an empty string!“
- Възрастта трябва да число по-голямо от 17, иначе да се генерира изключение със съобщението: „Age must be greater than 17 years!“
- Заплатата трябва да е положително число, иначе да се генерира изключение със съобщението: „Salary must be a positive number!“

Б) Класът Director наследява ClubMember и съдържа собствени полета:

- directorType (вид директор) – низ – една от думите executive, technical или sports

Класът трябва да притежава следните методи:

- **конструктор**, който задава всички стойности на характеристиките;
- **предефиниран метода** Info (за C#) и info (за Java), така че да се извежда на стандартния изход информацията в следния формат:

На първия ред: <вид директор> director: <име> <фамилия>

На втория ред: salary: <заплата> lv

На третия ред: age: <възраст> years

Заплатата да бъде форматирана до втория знак след десетичния разделител.

В) Класът Coach наследява ClubMember и съдържа собствени характеристики:

- coachType (вид треньор) – низ – една от думите: head, assistant или goalkeeper
- contractLength (продължителност на договора) – цяло число

Класът трябва да притежава следните методи:

- **конструктор**, който задава всички стойности на характеристиките;
- **предефиниран метода Info** (за C#) и **info** (за Java), така че да се извежда на стандартния изход информацията в следния формат:

На първия ред: <вид треньор> coach: <име> <фамилия>

На втория ред: salary: <заплата> lv

На третия ред: age: <възраст> years

Заплатата да бъде форматирана до втория знак след десетичния разделител.

Г) Класът FootballPlayer наследява ClubMember и съдържа собствени характеристики:

- **position** (позиция) – низ – една от думите: forward, midfielder, defender, goalkeeper
- **contractLength** (продължителност на договора) – цяло число
- **matches** (брой мачове) – цяло число
- **goals** (брой голове) – цяло число
- **assist** (брой асистенции) – цяло число

Класът трябва да притежава следните методи:

- **конструктор**, който задава всички стойности на характеристиките;
- **предефиниран метода Info** (за C#) и **info** (заJava), така че да се изведе на стандартния изход информацията в следния формат:

На първия ред: <име> <фамилия> - <позиция>

На втория ред: salary: <заплата> lv

На третия ред: age: <възраст> years

На четвъртия ред: <брой голове> goals and <брой асистенции> assists in <брой мачове> matches

Д) Приложението трябва да прочете от текстов файл **input.txt** данните за членовете на футболен клуб, като информацията за всеки член на клуба е записана на един ред. Приложението трябва да извежда на стандартният изход информацията за всеки един от членовете на клуба, сортирани по години в нарастващ ред, като информацията за всеки член на клуба се отделя с един ред, съдържащ двадесет звездички. Накрая на отделни редове да се изведат следните две съобщения:

The person with the highest salary in the club is <име> <фамилия> with <заплата> lv salary.

The team's top scorer is <име> <фамилия> with <брой голове> goals.

Приложението трябва да прихваща и обработва изключенията.

Пример с използване на ресурсния файл с име input.txt:

Изход:

Petar Petrov - forward

salary: 9000.00 lv

age: 18 years

3 goals and 0 assists in 10 matches

Ivan Ivanov - forward

salary: 10000.00 lv

age: 22 years

25 goals and 10 assists in 50 matches

Kaloyan Kaloyanov - defender

salary: 6000.00 lv

age: 24 years

5 goals and 2 assists in 80 matches

Georgi Georgiev - midfielder

salary: 11000.00 lv

age: 27 years

19 goals and 33 assists in 100 matches

Dimitar Dimitrov - goalkeeper

salary: 7500.00 lv

age: 35 years

0 goals and 0 assists in 300 matches

assistant coach: Plamen Kirov

salary: 3000.00 lv

age: 41 years

sports director: Biser Biserov

salary: 18000.00 lv

age: 48 years

head coach: Gari Plamenov

salary: 4000.00 lv

age: 51 years

executive director: Daniel Danielov

salary: 25000.00 lv

age: 55 years

technical director: Grigor Grigorov

salary: 20000.00 lv

age: 61 years

The person with the highest salary in the club is Daniel Danielov with 25000.00 lv salary.

The team's top scorer is Ivan Ivanov with 25 goals.

Прикачете в изпитната система архив с име inf_23.08.2024_ zad28.zip, съдържащ файловете с Вашите решения.