

**ДЪРЖАВЕН ЗРЕЛОСТЕН ИЗПИТ ПО
ИНФОРМАТИКА**

25 август 2023 г.

ПРОФИЛИРАНА ПОДГОТОВКА

ВАРИАНТ 2

ЧАСТ 1 (Време за работа: 90 минути)

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

1. В коя от следните двойки първият обект е във връзка от тип „е“ (is-a) с втория?

- А) превозно средство – автобус
- Б) двигател – кола
- В) превозно средство – двигател
- Г) кола – превозно средство

2. Информационна система се нарича:

- А) Информация, оформена по подходящ начин за съхранение в компютрите.
- Б) Съвкупност от данни, съхранени и организирани по определен начин, така че да могат да бъдат ефективно обработвани с компютър.
- В) Система за събиране, обработка и предоставяне на информация с цел задоволяване на специфичните нужди на определена компания.
- Г) Сведения за обектите и явленията от обкръжаващата ни среда, техните параметри, свойства и състояния, които се предават от един източник на друг с цел комуникация, разбиране и вземане на решения.

3. През кой етап от жизнения цикъл на една информационна система се определят предназначението, основните функции и изискванията към разработвания софтуерен продукт?

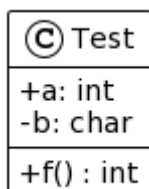
- А) Проектиране
- Б) Тестване
- В) Анализ на системата
- Г) Разработка

4. Посочете какво трябва да се добави или премахне в описанието на класовете, за да се получи коректен код.

```
public abstract class BaseClass
{
    abstract void Addition(double liters);
}
public class DeviseeClass:BaseClass
{
    public double Quantity { get; set; }
    public override void Addition(double liters)
    {
        this.Quantity = liters * 0.20;
    }
}
```

- А) Служебната дума `override` трябва да се премахне.
- Б) Кодът е верен и нищо не трябва да се премахва или добавя.
- В) Трябва в описанието на класа `BaseClass` пред метода да се добави `private`.
- Г) Трябва в описанието на класа `BaseClass` пред метода да се добави `public`.

5. Дадена е UML диаграма на клас. Вярно ли е, че тя съответства на дадения програмен код?



```
class Test {
    public int a;
    private char b;
    public int f() { return 0; }
}
```

- А) Не, защото функцията **f** трябва да е от тип **void** и да има параметър от тип **int**.
- Б) Да, защото типа на данните и модификаторите за достъп са правилни.
- В) Не, защото променливата **b** трябва да бъде с модификатор за достъп **protected**.
- Г) Не, защото **a** и **b** трябва да бъдат с модификатор за достъп **public**, а **f** – с **protected**.

6. Отбележете кое от твърденията за модификаторите за достъп до елементите на класа НЕ е вярно:

А) Модификатори за достъп могат да се използват пред следните елементи на класа: полета/свойства и методи на класа.

Б) **public**, **protected**, **private** и **readonly** (за C#) / **final** (за Java) са модификатори за достъп.

В) Нивото на достъп, което налага най-много ограничения е **private**.

Г) Модификаторите за достъп ограничават или позволяват видимост до елементите, пред които са поставени.

7. Какво ще се изведе на стандартния изход, при изпълнението на следния програмен фрагмент?

C#

```
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("]");
```

Java

```
List<Integer> numbers =
    new ArrayList<>(Arrays.asList(10, 20, 30, 40, 35, 25, 15));
int n = numbers.size();
for (int i = 0; i < n/2; i++) {
    int p = numbers.get(i);
    numbers.set(i, numbers.get(n-1-i));
    numbers.set(n-1-i, p);
}
System.out.println(String.join(", ", numbers.toString()));
```

А) [15, 25, 35, 40, 30, 20, 10]

Б) [10, 15, 20, 25, 30, 35, 40]

В) [35, 25, 15, 40, 10, 20, 30]

Г) [40, 35, 30, 25, 20, 15, 10]

8. Посочете кой от изброените графични компоненти е най-подходящо да се използва, когато на потребителя на приложението, трябва да се даде възможност за избор на точно една от няколко възможни опции.

- А) кутийка за отметки (Checkbox);
- Б) етикет (Label);
- В) текстово поле (Textfield);
- Г) радио бутон (Radio button);

9. В коя от посочените двойки връзката между елементите на двойката е от тип много към много?

- А) апартамент – стая
- Б) животно – крайник
- В) ученик – състезание
- Г) шкаф – дреха

10. За да може да се използва колекцията `List<T>` коя стандартна библиотека трябва да се включи:

C#	Java
А) System.Collections	А) java.util.ArrayList
Б) System.Windows.Forms	Б) javax.swing.ButtonGroup
В) System.Collections.Generic	В) java.util.List
Г) System.Linq	Г) java.util.LinkedList

11. Какъв ще е резултатът от изпълнението на програмния код, ако за стойност на `studentGrade` се въведе 6.01?

```
C#
class MyException:Exception{ }
class DZI
{
    static void Main(string[] args)
    {
        MyException myException = new MyException();
        double studentGrade = double.Parse(Console.ReadLine());
        List<double> studentGrades = new List<double>();

        try
```

```

    {
        if (studentGrade < 2 || studentGrade > 6) throw myException;
        else studentGrades.Add(studentGrade);
    }
    catch (MyException)
    {
        Console.WriteLine("Invalid input data.");
    }
}
}

```

Java

```

public class MyException extends Exception {
}
public class DZI {
    public static void main(String[] args) {
        MyException myException = new MyException();
        Scanner scanner = new Scanner(System.in);
        List<Double> studentGrades = new ArrayList<>();
        try {
            double studentGrade = Double.parseDouble(scanner.nextLine());
            if (studentGrade < 2 || studentGrade > 6) throw myException;
            else studentGrades.add(studentGrade);
        } catch (MyException e) {
            System.out.println("Invalid input data.");
        }
    }
}
}

```

- A) Ще се добави нова стойност в списъка `studentGrades`, но на стандартния изход няма да се изведе нищо.
- Б) Програмата няма да се компилира поради грешка в кода.
- В) На стандартния изход ще се изведе съобщението „Invalid input data.“.
- Г) На стандартния изход ще се изведе съобщението: „System.IndexOutOfRangeException: Index was outside the bounds of the array.“.

12. Какво ще изведе следния програмен фрагмент:

C#

```
var alist = new ArrayList {4, 3, "Pepo"};
string[] names = {"Koko", "Emi"};
alist.AddRange(names);
alist.Insert(1, 3);
alist.RemoveAt(4);
alist.Remove(3);
foreach (var elem in alist)
{
    Console.Write($"{elem} ");
}
```

Java

```
ArrayList<Object> alist = new ArrayList<> (Arrays.asList(4,3,"Pepo"));
String[] names = {"Koko", "Emi"};
alist.addAll(Arrays.asList(names));
alist.add(1, 3);
alist.remove(4);
alist.remove((Object)3);
for (Object elem : alist) {
    System.out.print(elem + " ");
}
```

A) 4 3 3 Emi

Б) 4 3 Pepo Emi

В) 3 Pepo Koko Emi

Г) няма да се компилира, защото не може в ArrayList да има данни от различен тип

13. Дадени са два файла с имена `old.txt` и `new.txt`, съдържащи пет реда със следното съдържание:

old.txt	new.txt
Hello, John	Hello, John
I will be in Sofia tomorrow.	My flight was delayed.
I have the documents.	I have the documents.
Looking forward to seeing you!	I will call you when I arrive.
Best, Mary	Best, Mary

Какво ще е съдържанието на файла с име `diff.txt` след изпълнението на дадения програмен фрагмент?

C#

```
StreamReader readerOld = new StreamReader("old.txt");
StreamReader readerNew = new StreamReader("new.txt");
```

```

StreamWriter writer = new StreamWriter("diff.txt");
while (readerOld.Peek() >= 0)
{
    String lineOld = readerOld.ReadLine();
    String lineNew = readerNew.ReadLine();
    if (lineOld.CompareTo(lineNew) == 0) writer.WriteLine(lineOld);
    else
    {
        writer.WriteLine("- " + lineOld);
        writer.WriteLine("+ " + lineNew);
    }
}
readerOld.Close();
readerNew.Close();
writer.Close();

```

Java

```

Scanner scannerOld = new Scanner(new FileInputStream("old.txt"));
Scanner scannerNew = new Scanner(new
FileInputStream("new.txt"));
try (PrintWriter writer = new PrintWriter("diff.txt")) {
    while (scannerOld.hasNextLine()) {
        String lineOld = scannerOld.nextLine();
        String lineNew = scannerNew.nextLine();
        if (lineOld.compareTo(lineNew) == 0)
            writer.println(lineOld);
        else {
            writer.println("- " + lineOld);
            writer.println("+ " + lineNew);
        }
    }
}

```

A)

```

+ Hello, John
I will be in Sofia tomorrow.
My flight was delayed.
- I have the documents.
Looking forward to seeing you!
I will call you when I arrive.
+ Best, Mary

```

B)

```

Hello, John
- I will be in Sofia tomorrow.
+ My flight was delayed.
I have the documents.
- Looking forward to seeing you!
+ I will call you when I arrive.
Best, Mary

```

B)

```

- Hello, John
+ Hello, John
I will be in Sofia tomorrow.
- I have the documents.
+ I have the documents.
Looking forward to seeing you!
- Best, Mary
+ Best, Mary

```

Γ)

```

Hello, John
- My flight was delayed.
+ I will be in Sofia tomorrow.
I have the documents.
- I will call you when I arrive.
+ Looking forward to seeing you!
Best, Mary

```

14. Посочете вярното твърдение за публичен статичен член на клас:

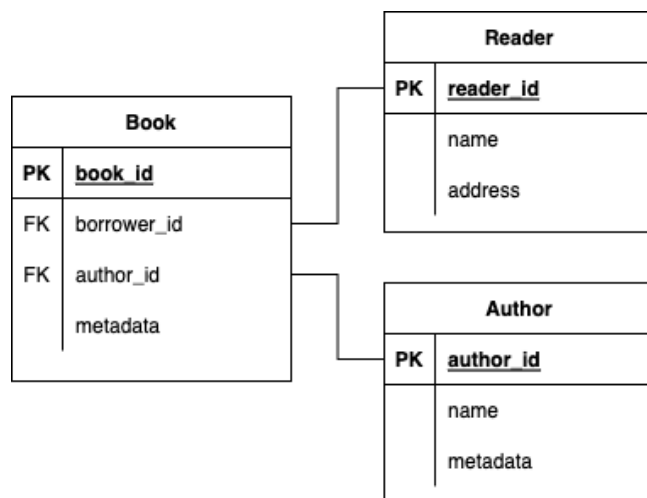
- А) получава стойност само при декларирането си, която след това не може да се променя;
- Б) достъпен е само от обекти на класа, в който е деклариран;
- В) той е уникален за целия клас и може да се достъпва чрез името на класа;
- Г) може да се дефинира само в статичен клас.

15. При коя от следните структури от данни НЕ може да се достъпи пряко елемент на произволна позиция?

- А) Опашка
- Б) Символен низ (string)
- В) Масив
- Г) Списък

16. Дадената диаграма описва база от данни с книги в библиотека. Кое от изброените твърдения е вярно?

- А) В даден момент една книга може да бъде заета от повече от един читател.
- Б) Дадена книга може да има само един автор.
- В) В даден момент даден читател може да заеме книги само от един и същи автор
- Г) В даден момент книги на даден автор могат да бъдат заети само от един читател.



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

17. Дефиниран е двумерен масив arr (m x m) от цели числа.

C#	Java
<code>int[,] arr = new int[m, m];</code>	<code>int[][] arr = new int[m][m];</code>

С дадения програмен код трябва да се намери броя на отрицателните елементи под главния диагонал на масива, но има изпуснати части отбелязани с номера (1),

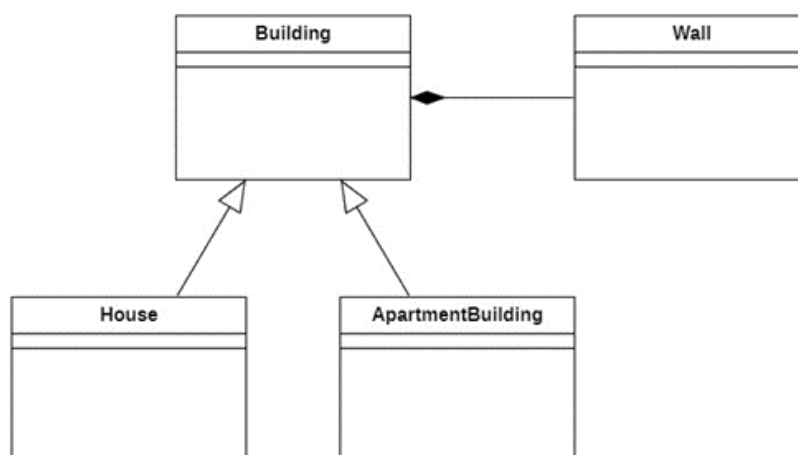
(2) и (3). В листа за отговори напишете (1), (2) и (3) и срещу тях запишете пропуснатите части от кода, така че да се получи верен програмен код.

```
int cnt = 0;
for (int i = 0; i < m; i++)
{
    for (int j = 0; ___(1)___; j++)
    {
        if (___(2)___)
        { ___(3)___ }
    }
}
```

18. Кое число ще върне като резултат извикването на дадения метод със стойности 2, 1 - `ack(2, 1)`?

```
static int ack(int m, int n) {
    if (m == 0) return n + 1;
    if (n == 0) return ack(m - 1, 1);
    return ack(m - 1, ack(m, n - 1));
}
```

19. В диаграмата са онагледени връзките между класовете `Building` (Сграда), `Wall` (Стена), `House` (Къща) и `ApartmentBuilding` (Жилищен блок).



Определете дали типа на връзката е “има” (has a) или “е” (is a) между посочените класове.

В листа за отговори запишете (1), (2) и (3) и срещу тях напишете:

- (1) връзката между `Building` и `Wall`
- (2) връзката между `House` и `Building`
- (3) връзката между `ApartmentBuilding` и `Wall`

20. Дадена е таблицата Fee.

admission_no	course	amount_paid
3354	Java	2000
7555	C#	1800
4321	SQL	1600
8345	C++	1200
5112	Python	2200

В листа за отговори запишете (1), (2) и (3) и срещу тях напишете **КЛЮЧОВИТЕ ДУМИ** за трите заявки, с които се:

- 1) добавя курс с име Scratch, имащ admission_no 2727 и amount_paid 1500;
- 2) променя стойността на amount_paid на курса C# на 2000;
- 3) изтриват всички курсове със стойност на amount_paid поне 2000.

21. В листа за отговори запишете с кратък текст (1 - 2 изречения) какъв е резултатът от извикването на метода Proceed(), като е гарантирано, че при извикване на метода $a \leq b$.

```
static int Proceed(int a, int b)
{
    int count = 0;
    for (int i = a; i <= b; i++)
    {
        int num = i;
        while (num > 0)
        {
            int d = num % 10;
            num /= 10;
            if (d % 2 != 0) continue;
            count++;
        }
    }
    return count;
}
```

22. Дадена е таблицата cars със следните данни:

year	make	model	price
2014	B	F1	15000
2001	B	EA2	5000
2022	H	A012	40000
1999	T	ABC	3000
2000	T	MX	10000

Какъв ще бъде резултатът от изпълнението на дадената SQL заявка?

В листа за отговори запишете отговора в табличен вид.

SQL	MS Access
<pre>SELECT COUNT(*) AS cars_count, MIN(year) AS first_year, SUM(price) AS total_price FROM cars WHERE model NOT LIKE 'A%' OR make = 'H';</pre>	<pre>SELECT COUNT(*) AS cars_count, MIN(year) AS first_year, SUM(price) AS total_price FROM cars WHERE model NOT LIKE "A*" OR make = "H";</pre>

23. В листа за отговори запишете какъв резултат ще се изведе на стандартния изход след изпълнение на следния програмен фрагмент:

C#
<pre>string text = "Държавен зрелостен изпит по информатика 2023"; Console.WriteLine(text.Substring(28)); Console.WriteLine(text.Remove(9,10)); var output = text.Split(' ').ToList(); Console.WriteLine(string.Join("\n",output));</pre>
Java
<pre>String text = "Държавен зрелостен изпит по информатика 2023"; System.out.println(text.substring(28)); System.out.println(text.substring(0, 9) + text.substring(19)); List<String> output = Arrays.asList(text.split(" ")); System.out.println(String.join("\n", output));</pre>

24. Даденият метод трябва да извежда средноаритметичното на елементите на едномерен масив от цели числа, след нулевия елемент, които са кратни на индексите си. В програмния код има няколко грешки. В листа за отговори за всяка грешка посочете номера на реда, в който се намира и напишете верния код.

C#
<pre>1. static void Avg(int[] a) 2. { int cnt = 0, s = 0; 3. for (int i = 1; i < a.Length(); i++) 4. { if (a[i] % i = 0) 5. { s + a[i]; 6. cnt++; 7. } 8. } 9. double avg = (cnt==0) ? 0 : (double) s / cnt; 10. Console.Write(\$"{avg:F2}"); 11. }</pre>
Java
<pre>1. static void avg(int[] a){ 2. int cnt = 0, s = 0; 3. for (int i = 1; i < a.length(); i++){ 4. if (a[i] % i = 0) { 5. s + a[i]; 6. cnt++; 7. } 8. } 9. double avg = (cnt==0) ? 0 : (double) s / cnt; 10. System.out.printf("%.2f", avg); 11. }</pre>

ДЪРЖАВЕН ЗРЕЛОСТЕН ИЗПИТ ПО

ИНФОРМАТИКА

25 август 2023 г.

ПРОФИЛИРАНА ПОДГОТОВКА

ВАРИАНТ 2

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 `Zad25`, в който се прочита от стандартния вход естествено число n , $1 \leq n \leq 10000000$. Програмата проверява и извежда на стандартния изход съобщението: „<число> is a palindrome“, ако въведеното число е палиндром. Ако въведеното число не е палиндром да се извежда съобщението: „<число> is NOT a palindrome“. Направете необходимата валидация и обработка на изключение. Ако въведената данна не е валидна да се извежда съобщението „Incorrectly entered number“.

Упътване: Едно число е палиндром, ако се чете по един и същ начин от ляво на дясно и от дясно на ляво.

*Забележки: Приемат се и решения с графичен потребителски интерфейс (ГПИ), в които числото n се въвежда в текстово поле, а резултатът се извежда в етикет или нередактируемо текстово поле.

Пример:

Вход	Изход
12321	12321 is a palindrome
123	123 is NOT a palindrome
text	Incorrectly entered number
-2002	Incorrectly entered number

Прикачете в изпитната система архив с име `zad25.zip`, съдържащ файловете с Вашите решения.

26. Създайте конзолен проект с име Zad26, който прочита от стандартния вход цяло число N ($3 \leq N \leq 10000$). От следващите N реда програмата трябва да прочете по едно реално число от интервала [-100, 100]. Числата представляват резултатите в точки, получени от зрелостници, явили се на матура по информатика. Резултатите за валидните работи на учениците са положителни числа, за неявилите се или недопуснатите са 0, а за анулираните - отрицателни числа, според причината за анулиране.

Трябва да се направят справки за: броя на валидните предадени работи, минималната разлика между различните точки и броя на лауреатите (всички ученици с точки, не по-малки от първите 3 най-високи резултата). Резултатът от направените справки трябва да се изведе на екрана.

Проекта трябва да съдържа посочените по-долу методи, които да се извикват от главната функция. За всеки от тези методи е фиксирано само името, а типа на връщания резултат, броят и типа на параметрите, имате свободата да определите Вие.

- 1) ReadPoints() – прочита резултатите на учениците от стандартния вход и съхранява в колекция само точките за валидните работи;**
- 2) MinDpoints() – връща като резултат минималната разлика между различните точки за валидните работи;**
- 3) Laureates() – връща като резултат броя зрелостниците с точки, не по-малко от първите 3 най-високи резултати;**

Като се използват тези методи, приложението трябва да намери и изведе на стандартния изход резултатите, като спазва следния формат:

- на първия ред – записва се броя на валидните работи (с положителни точки) във формат: „**valid works - <число>**”
- на втория ред – записва се минималната разлика между различните точки за валидните работи във формат: „**minimal difference - <число> p.**“ Числото да е закръглено до третия знак след десетичната запетая.
- на третия ред – записва се брой лауреати във формат: „**laureates - <число>**“.

Ограничения: Точките са в интервала [-100, 100]. Гарантирано е, че има поне 3 различни стойности за точки на валидни работи.

Пример:

Вход:	Изход:
15 75.125 86.257 85.235 99.9 -5 0 94.235 -2 90.135 81.145	valid works - 11 minimal difference - 1.022 p. laureates - 4

0	
86.257	
97.145	
86.257	
97.145	

Прикачете в изпитната система архив с име zad26.zip, съдържащ файловете с Вашите решения.

27. Създайте база от данни с име music, съдържаща информация за няколко изпълнители.

А) Създайте таблица singers, съхраняваща информация за певци. Таблицата трябва да има следните колони:

- id – номер на певица (цяло число), първичен ключ;
- name - име на певица (текст);
- songs - брой песни (цяло число);
- rank - номер в класацията за най-добър певец (цяло число);
- networth - нетно състояние в лева (цяло число).

Б) Добавете в таблицата следните кортежи (данните за таблицата са в ресурсния файл с име resources.txt):

id	name	songs	rank	networth
1	Ivan Ivanov	50	1	1000000
2	Maria Ivanova	55	3	900000
3	Georgi Georgiev	20	4	800000
4	Gergana Petrova	55	2	1000000
5	Boris Borisov	20	5	900000

В) Напишете заявка, която извежда ранга и имената на тримата най-добри певци (певците с ранг 1, 2 и 3 в класацията), подредени във възходящ ред по ранг.

Г) Напишете заявка, която извежда общия брой песни на всички изпълнители, както и средното им нетно състояние в евро (в таблицата нетното състояние е дадено в лева, а 1 € = 1.95583 лв).

Д) Напишете заявка, която увеличава с 10% нетното състояние на певците с номера между 2 и 4 (включително).

Прикачете в изпитната система архив с име 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. Създайте приложение с име Zad28, което имплементира следните класове с описаните атрибути:

Клас Car с членове:

- brand - модел (низ)
- hPower – мощност в конски сили (цяло число)
- конструктор, който задава стойности на характеристиките;
- метод за предефиниране на метода ToString (за C#) / toString (за Java) за клас, така че да връща низа: „<модел>, <мощност>“

Клас Pilot – наследник на Person с членове:

- name – име на участника (низ)
- age – възраст (цяло число)
- car (обект от клас Car)
- category – категория за шофьорска книжка (низ)
- конструктор, който задава стойности на характеристиките;
- метод за предефиниране на метода ToString (за C#) / toString (за Java) за клас, така че да връща низа:
„<име>, <възраст>, <категория>, <модел>, <мощност>“

Клас Rally с членове:

- name - име на ралито (низ),
- year - година на провеждане (цяло число)
- pilots - списък от участници от клас Pilot
- конструктор, който задава стойности на характеристиките name и year и създава празен списък от участници.;
- метод за добавяне на участник
- метод за извеждане на стандартния изход на информация за ралито във вида:

На първия ред да се извежда текста:

„Rally: < име на ралито> - < година на провеждане >”

Следващите редове съдържат данни за участниците в ралито. Всеки ред съдържа данни за един участник във формат:

„<име>, <възраст>, <категория>, <модел>, <мощност>“

Приложението трябва да реализира функционалности, които се управляват с еднобуквени команди a, v и q, които потребителят да може да избира от меню в следния вид: [a]dd [v]iew [q]uit

Като опциите от менюто служат за:

- add – добавя данни за пилот;
- view – извежда информация за участниците;
- quit –излиза от приложението;

Изисквания към приложението:

1. При стартиране на приложението, данните за името и година на ралито се прочитат съответно от първия и от втория ред на стандартния вход и се съхраняват в обект от тип Rally.
2. При добавяне на пилот, всички данни се съхраняват в създадения обект от тип Rally.
3. При въвеждане на данни да се спазва формата даден в примерите.

Пример:

Примерна работа на програмата (данните въвеждани от потребителя са оградени в правоъгълник) Входни данни за примера:

```
Pilots_Rally
2023
```

```
[a]dd [v]iew [q]uit
a
Pilot name: Martin Manchev
Age: 23
Category (A|B|C): A
Car model: BMW
Car power (hp): 99
```

```
[a]dd [v]iew [q]uit
a
Pilot name: Rado Lilov
Age: 23
Category (A|B|C): B
Car model: Nissan
Car power (hp): 100
```

```
[a]dd [v]iew [q]uit
v
Rally: Pilots_Rally - 2023
Martin Manchev,23,A,BMW,99
Rado Lilov,23,B,Nissan,100
```

```
[a]dd [v]iew [d]elete [q]uit
q
```

```
Pilots_Rally
2023
a
Martin Manchev
23
A
BMW
99
a
Rado Lilov
23
B
Nissan
100
v
q
```

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

МИНИСТЕРСТВО НА ОБРАЗОВАНИЕТО И НАУКАТА

ДЪРЖАВЕН ЗРЕЛОСТЕН ИЗПИТ ПО

ИНФОРМАТИКА

25 август 2023 г.

ПРОФИЛИРАНА ПОДГОТОВКА

ВАРИАНТ 2

Задача от 1. до 16. Ключ с верните отговори

Въпрос №	Верен отговор	Брой точки
1.	Г	1
2.	В	1
3.	В	1
4.	Г	1
5.	Б	1
6.	Б	1
7.	А	1
8.	Г	1
9.	В	1
10.	В	1
11.	В	1
12.	Б	1
13.	Б	1
14.	В	1
15.	А	1
16.	Б	1

Задача 17. – 3 точки

(1) $j < i$

(2) $arr[i, j] < 0$ (за C#)

$arr[i][j] < 0$ (за Java)

или

(1) $j < m$

(2) $arr[i, j] < 0 \ \&\& \ j < i$ (за C#)

$arr[i][j] < 0 \ \&\& \ j < i$ (за Java)

(3) $cnt++$; или $++cnt$; или $cnt += 1$; или $cnt = cnt + 1$;

Задача 18. – 3 точки

5

Задача 19. – 3 точки

(1) има (has a)

(2) е (is a)

(3) има (has a)

Задача 20. – 3 точки

1) INSERT INTO Fee

VALUES

Или

INSERT INTO ... VALUES ...

2) UPDATE Fee SET ...

WHERE ...

Или

UPDATE ... SET ... WHERE ...

3) DELETE FROM Fee

WHERE ...

Или

DELETE FROM ... WHERE ...

Задача 21. – 3 точки

Методът връща като резултат броят четни цифри на положителните числата от интервала [a ; b]. Ако a и b са цели неположителни числа, методът връща резултат нула.

Задача 22. – 3 точки

cars_count	first_year	total_price
4	2000	70000

Задача 23. – 3 точки

информатика 2023

Държавен изпит по информатика 2023

Държавен

зрелостен

изпит

по

информатика

2023

Задача 24. – 3 точки

ред 3 – грешен синтаксис за взимане на дължината на масива, посочено е като метод, вместо като свойство. Правилният код за реда е:

за C# `for (int i = 1; i < a.Length; i++)`

за Java `for (int i = 1; i < a.length; i++)`

ред 4 – грешна проверка дали `a[i]` се дели на `i`. Правилният код за реда е:

`if (a[i] % i == 0)`

ред 5 – липсва оператор за присвояване. Правилният код за реда е:

`s += a[i];` или `s = s + a[i];`

Задача 25. – 10 точки

Примерно решение

C#

```
using System;
using System.Collections.Generic;

namespace zad25
{
    internal class Program
    {
        static void Main()
        {
            int number;
            int originalNumber;
            int n; // дължината на списъка
            bool check = true; // проверка за палиндром
            List<int> digits = new List<int>();

            try
            {
                number = int.Parse(Console.ReadLine());
                //валидация
                if (number < 1 || number > 10000000) throw new Exception();

                originalNumber = number;
                while (number > 0) // отделяне на цифрите
                {
                    digits.Add(number % 10);
                    number /= 10;
                }

                n = digits.Count;
                for (int i = 0 ; i < n / 2 ; i++) // проверка за палиндром
                {
                    if (digits[i] != digits[n - i - 1])
                    {
                        check = false;
                        break;
                    }
                }

                if (check)
                {
                    Console.WriteLine(originalNumber + " is a palindrome");
                }
                else
                {
                    Console.WriteLine(originalNumber + " is NOT a palindrome");
                }
            }
            catch (Exception )
            {
                Console.Error.WriteLine("Incorrectly entered number");
            }
        }
    }
}
```

Java

```

package dzi;

import java.util.ArrayList;
import java.util.Scanner;

public class zad25 {

    public static void main(String[] args) {
        int number;
        int originalNumber;
        int n; // дължината на списъка
        boolean check = true; // отметка за палиндром
        ArrayList<Integer> digits = new ArrayList<>();
        try (Scanner input = new Scanner(System.in)) {
            try {
                number = input.nextInt();
                // валидация
                if (number < 1 || number > 10000000)
                    throw new Exception();
                originalNumber = number;
                // отделяне на цифрите и записване в списък
                while (number > 0) {
                    digits.add(number % 10);
                    number /= 10;
                }

                n = digits.size();
                for (int i = 0; i < n / 2; i++) { // проверка за палиндром
                    if (!digits.get(i).equals(digits.get(n - 1 - i))) {
                        check = false;
                        break;
                    }
                }

                if (check) {
                    System.out.printf("%d is a palindrome\n",
originalNumber);
                } else {
                    System.out.printf("%d is NOT a palindrome\n",
originalNumber);
                }
            } catch (Exception ex) {
                System.err.println("Incorrectly entered number");
            }
        }
    }
}

```

Задача 26 – 15 точки

Примерно решение

C#

```

using System;
using System.Collections.Generic;

namespace zad26
{
    internal class Program
    {
        static List<double> ReadPoints()

```

```

{
    List<double> points = new List<double>();
    int n = int.Parse(Console.ReadLine());
    double point;
    for (int i = 0 ; i < n ; i++)
    {
        point = double.Parse(Console.ReadLine());
        if (point > 0)
        {
            points.Add(point);
        }
    }
    return points;
}
static double minDpoints(List<double> points)
{
    double minD = 101;
    for (int i = 1 ; i < points.Count ; i++)
    {
        if (points[i].Equals(points[i - 1])) continue;
        double d = points[i - 1] - points[i];
        if (d < minD) minD = d;
    }
    return minD;
}
static int Laureates(List<double> points)
{
    int countLaureates = 1, cntTo3 = 1;
    for (int i = 1 ; i < points.Count ; i++)
    {
        if (points[i] != points[i - 1])
        {
            cntTo3++;
            if (cntTo3 > 3) break;
        }
        countLaureates++;
    }
    return countLaureates;
}
static void Main()
{
    // прочитане на точките и записване в колекция на валидните работи
    List<double> points = ReadPoints();

    //сортиране на данните в намаляващ ред
    points.Sort((a, b) => b.CompareTo(a));

    // минимална разлика
    double d = minDpoints(points);

    // намиране на броя на лауреатите
    int cntLaureates = Laureates(points);

    // извеждане на резултатите
    Console.WriteLine($"valid works - {points.Count}");
    Console.WriteLine($"minimal difference - {d:F3} p.");
    Console.WriteLine($"laureates - {cntLaureates}");
}
}
}

```

Java

```
package dzi;
```

```
import java.util.ArrayList;
import java.util.Collections;
import java.util.Scanner;
```

```

public class zad26 {

    static ArrayList<Double> readPoints() {
        ArrayList<Double> points = new ArrayList<>();
        Scanner sc = new Scanner(System.in);
        int n = Integer.parseInt(sc.nextLine());
        double point;
        for (int i = 0; i < n; i++) {
            point = Double.parseDouble(sc.nextLine());
            if (point > 0) {
                points.add(point);
            }
        }
        return points;
    }

    static int Laureates(ArrayList<Double> all) {
        int countAll = 1, cntTo3 = 1;
        for (int i = 1; i < all.size(); i++) {
            if (!(all.get(i).equals(all.get(i - 1)))) {
                cntTo3++;
                if (cntTo3 > 3)
                    break;
            }
            countAll++;
        }
        return countAll;
    }

    static double minDpoints(ArrayList<Double> li) {
        double minD = 101;
        double td;
        for (int i = 1; i < li.size(); i++) {
            if (li.get(i).equals(li.get(i - 1)))
                continue;
            td = li.get(i - 1) - li.get(i);
            if (td < minD) {
                minD = td;
            }
        }
        return minD;
    }

    public static void main(String[] args) {
        // прочитане на поредицата от числа
        ArrayList<Double> points = readPoints();

        // сортиране на данните в намаляващ ред
        Collections.sort(points);
        Collections.reverse(points);

        // минимална разлика
        double d = minDpoints(points);

        // намиране на броя на лауреатите
        int cntLaureates = Laureates(points);

        // извеждане на резултатите
        System.out.println("valid works - " + points.size());
        System.out.println(String.format("minimal difference - %.3f p.", d));
        System.out.println("laureates - " + cntLaureates);
    }
}

```



```
}  
}
```

Задача 27 – 15 точки

Примерно решение

```
-- А)  
CREATE DATABASE music;  
  
CREATE TABLE singers(  
    id INT PRIMARY KEY,  
    name NVARCHAR(100),  
    songs INT,  
    rank INT,  
    networth INT  
);  
  
-- Б)  
INSERT INTO singers(id, name, songs, rank, networth)  
VALUES (1, 'Ivan Ivanov', 50, 1, 1000000),  
(2, 'Maria Ivanova', 45, 3, 900000),  
(3, 'Georgi Georgiev', 20, 4, 800000),  
(4, 'Gergana Petrova', 55, 2, 1000000),  
(5, 'Boris Borisov', 35, 5, 900000);  
  
-- В)  
SELECT rank, name  
FROM singers  
WHERE rank <= 3  
ORDER BY rank;  
  
rank      name  
-----  
1         Ivan Ivanov  
2         Gergana Petrova  
3         Maria Ivanova  
  
-- Г)  
SELECT COUNT(*), AVG(networth) / 1.95583  
FROM singers;  
  
-----  
5         470388.5307005  
  
-- Д)  
UPDATE singers  
SET networth = networth * 1.1  
WHERE ID BETWEEN 2 AND 4
```

След изпълнение на заявката таблицата ще има следното съдържание:

```
-----  
1         Ivan Ivanov          50         1         1000000  
2         Maria Ivanova       55         3         990000  
3         Georgi Georgiev     20         4         880000  
4         Gergana Petrova     55         2         1100000  
5         Boris Borisov      20         5         900000
```

Задача 28 – 20 точки

Примерно решение

C#

```
using System;
using System.Collections.Generic;

namespace zad28
{
    class Pilot
    {
        private string name;
        int age;
        private string category;
        private Car carp;
        public Pilot(string name, int age, string categ, Car car)
        {
            this.name = name;
            this.age = age;
            this.category = categ;
            this.carp = car;
        }
        public override string ToString()
        {
            return $"{name},{age},{category},{carp}";
        }
    }

    class Car
    {
        private string model;
        private int hPower;
        public Car(string model, int hPower)
        {
            this.model = model;
            this.hPower = hPower;
        }
        public override string ToString()
        {
            return $"{model},{hPower}";
        }
    }

    class Rally
    {
        private string rName;
        private int rYear;
        private List<Pilot> pilots;
        public Rally(string name, int year)
        {
            this.rName = name;
            this.rYear = year;
            pilots = new List<Pilot>();
        }
        public void AddPilot()
        {
            Console.Write("Pilot name: ");
            string name = Console.ReadLine();
            Console.Write("Age: ");
            int age = int.Parse(Console.ReadLine());
            Console.Write("Category (A|B|C): ");
            string categ = Console.ReadLine();
            Console.Write("Car model: ");
        }
    }
}
```

```

        string model = Console.ReadLine();
        Console.Write("Car power (hp): ");
        int hpower = int.Parse(Console.ReadLine());
        Pilot pilot = new Pilot(name, age, categ, new Car(model, hpower));
        pilots.Add(pilot);
    }

    public void ViewAll()
    {
        Console.WriteLine($"Rally: {rName} - {rYear}");
        Console.WriteLine(string.Join(Environment.NewLine, pilots));
    }
}

class Zad28
{
    static void Main()
    {
        string rName = Console.ReadLine();
        int rYear = int.Parse(Console.ReadLine());
        Rally rally = new Rally(rName, rYear);
        while (true)
        {
            string inp;
            do
            {
                Console.WriteLine();
                Console.WriteLine("[a]dd [v]iew [q]uit");
                inp = Console.ReadLine();
            } while (inp != "a" && inp != "v" && inp != "q");

            if (inp == "q") { break; }

            switch (inp)
            {
                case "a":
                    rally.AddPilot();
                    break;
                case "v":
                    rally.ViewAll();
                    break;
            }
        }
    }
}

```

Java

```

package dzi;

import java.util.ArrayList;
import java.util.Scanner;

public class zad28 {

    private static Scanner input;

    public static void main(String[] args) {
        input = new Scanner(System.in);
        String name = input.nextLine();
        int year = Integer.parseInt(input.nextLine());

        Rally rally = new Rally(name, year);

        while (true) {

```

```

String inp;
do {
    System.out.println();
    System.out.println("[a]dd [v]iew [q]uit");
    inp = input.nextLine();
} while (!inp.equals("a") && !inp.equals("v") && !inp.equals("q"));

if (inp.equals("q")) {
    break;
}
switch (inp) {
case "a":
    rally.addPilot();
    break;
case "v":
    rally.ViewAll();
    break;
}
}
}

class Pilot {
    private String name;
    private int age;
    private Car carp;
    private String category;

    public Pilot(String name, int age, Car carp, String category) {
        this.name = name;
        this.age = age;
        this.carp = carp;
        this.category = category;
    }

    @Override
    public String toString() {
        return this.name + "," + this.age + "," + this.category + "," + this.carp;
    }
}

class Car {

    private String brand;
    private int hPower;

    public Car(String brand, int hPower) {
        this.brand = brand;
        this.hPower = hPower;
    }

    @Override
    public String toString() {
        return String.format("%s,%d", brand, hPower);
    }
}

class Rally {

    private String rName;
    private int rYear;
    private ArrayList<Pilot> pilots;

```

```

public Rally(String name, int year) {
    this.rName = name;
    this.rYear = year;
    this.pilots = new ArrayList<Pilot>();
}

public void addPilot() {
    Scanner sc = new Scanner(System.in);
    System.out.print("Pilot name: ");
    String name = sc.nextLine();
    System.out.print("Age: ");
    int age = Integer.parseInt(sc.nextLine());
    System.out.print("Category (A|B|C): ");
    String categ = sc.nextLine();
    System.out.print("Car model: ");
    String model = sc.nextLine();
    System.out.print("Car power (hp): ");
    int hpower = sc.nextInt();
    Pilot pilot = new Pilot(name, age, new Car(model, hpower), categ);
    pilots.add(pilot);
}

public void ViewAll() {
    System.out.println("Rally: " + this.rName + " - " + this.rYear);
    for (Pilot pilot : pilots) {
        System.out.println(pilot);
    }
}
}

```