

ИНФОРМАТИКА – ПРОФИЛИРАНА ПОДГОТОВКА

Държавен зрелостен изпит (ДЗИ) по информатика може да положите, ако във втори гимназиален етап сте изучавали учебния предмет за придобиване на профилирана подготовка. В този случай държавният зрелостен изпит включва учебно съдържание от задължителните модули на профилиращия учебен предмет, а цялата учебно-изпитната програма за изпита – времетраене, учебно съдържание, оценявани компетентности, общ брой и видове задачи, максимален брой точки и минимален праг за успешно полагане, е Приложение № 19 на Наредба № 7 от 11.08.2016 г. за профилираната подготовка.

ОБЩИ ПАРАМЕТРИ И ПРИМЕРНИ ЗАДАЧИ ЗА ДЗИ ПО ИНФОРМАТИКА (ПРОФИЛИРАНА ПОДГОТОВКА) ЗА УЧЕБНАТА 2021 - 2022 Г.

ДЗИ по информатика (профилирана подготовка) включва 28 задачи, както следва:

- 16 задачи с избираем отговор
- 8 задачи с кратък свободен отговор
- 4 задачи с разширен свободен отговор (практически задачи)

Всеки вид задачи е илюстриран с примери. Задачите проверяват конкретни компетентности, заложи в учебно-изпитната програма и в примерите те са посочени. Посочен е и максималният брой точки, които зрелостникът може да получи при вярно решение на съответната задача.

Следва да се има предвид обаче, че задачите не трябва да се възприемат като типови, които задължително ще се включват във всеки тестов вариант за ДЗИ. Формулировките на съответните задачи предполагат вариативност и няма да следват единен модел. Наред с това трябва да се отчита и че придобиването на една и съща компетентност може да се проверява през една или повече задачи от различни видове.

Максималният общ брой точки е 100.

Задачи с избираем отговор

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

C#	JAVA
<pre>for (int i = 15; i <= 18; i = i + 2) { listBox1.Items.Add(i); }</pre>	<pre>for (int i = 15; i <= 18; i = i + 2) { list.add(Integer.toString(i)); }</pre>

}		}	
а) 17;	б) Това е безкраен цикъл;	а) 17;	б) Това е безкраен цикъл;
в) 19;	г) 18;	в) 19;	г) 18;

Максимален брой точки: 1

Оценявани компетентности: *Познава синтаксиса на обектно-ориентиран език за програмиране. Познава основни операции над линейни абстрактни структури от данни.*

Определете вярното твърдение за следния фрагмент от програма:

`int[] arr = new int[5];`

- а) С този фрагмент се създава масив от пет цели числа.
- б) С този фрагмент се обхождат елементите на масив от цели числа.
- в) С този фрагмент се отпечатва петият елемент на масива.
- г) С този фрагмент се задава стойност на елемент на масив.

Максимален брой точки: 1

Оценявани компетентности: *Познава синтаксиса на обектно-ориентиран език за програмиране. Познава основни операции над линейни абстрактни структури от данни.*

Задачи с кратък свободен отговор

Дадена е следната рекурсивна статична функция:

C#	JAVA
<pre>public static void recFunc(int n) { if (n == 0) { return; } recFunc(n / 2); Console.WriteLine(n % 2); }</pre>	<pre>public static void recFunc(int n) { if (n == 0) { return; } recFunc(n / 2); System.out.print(n % 2); }</pre>

Запишете стойността, която ще бъде изведена на екрана след обръщението `recFunc(12)`.

Максимален брой точки: 3

Оценявани компетентности: *Проследява изпълнението на рекурсивен метод.*

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

C#	JAVA
<pre>using System; namespace Cls { class Program { static void Main(string[] args) { string result = ""; int[] numbers = { 4, 5, 6, 1, 8, 56, 55, 3, 2, 21 }; Console.WriteLine("Въведете две цели числа:"); try { int a = Convert.ToInt32(Console.ReadLine()); int b = Convert.ToInt32(Console.ReadLine()); int s = numbers[a] + numbers[b]; Console.WriteLine(s); } catch (FormatException) { result = "Вие не въведохте число"; Console.WriteLine(result); } catch (IndexOutOfRangeException) { result = "Вие излязохте извън границите на масива."; Console.WriteLine(result); } } } }</pre>	<pre>import java.util.Scanner; import java.lang.NumberFormatException; import java.lang.ArrayIndexOutOfBoundsException; public class Cls { public static void main(String[] args) { Scanner inp = new Scanner(System.in); String result = ""; int[] numbers = { 4, 5, 6, 1, 8, 56, 55, 3, 2, 21 }; System.out.println("Въведете две цели числа:"); try{ int a = Integer.parseInt(inp.nextLine()); int b = Integer.parseInt(inp.nextLine()); int s = numbers[a]+numbers[b]; System.out.println(s); } catch (NumberFormatException e){ result = "Вие не въведохте число"; System.out.println(result); } catch (ArrayIndexOutOfBoundsException e){ result = "Вие излязохте извън границите на масива."; System.out.println(result); } } }</pre>

Максимален брой точки: 3

Оценявани компетентности: Прихваща и обработка изключения със средствата на обектно-ориентиран език за програмиране.

Задачи с разширен свободен отговор (практическа задача)

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 – пробегът 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
```

Максимален брой точки: 15

Оценявани компетентности: *Прилага принципите на композиция и наследственост в обектно-ориентиран модел. Използва средствата на интегрирана среда за разработване, създаване, редактиране и тестване на програма. Използва форматиран текстов файл за вход и изход на данни. Реализира алгоритмично решение с използване на стандартни библиотеки и готови компоненти за търсене и сортиране над структури от данни.*