



Сборник със задачи

Ниво - Programming Basics

В този сборник могат да бъдат намерени задачи на ниво "Първи стъпки в програмирането". Задачите са съобразени с материала, който се изучава на това ниво. Задачите са перфектно упражнение, защото както всички програмисти знаят, писането е най-доброто упражнение след една лекция. В съдържанието на сборника е наблегнато на цикли, защото те се използват усилено и в по-горните нива и са най-важното нещо в това ниво. Проверката на задачите се изпълнява от система написана от автора на сборника.

1. Числата от 1 до 20

Напишете C# конзолна програма, която отпечатва числата от 1 до 20 на отделни редове на конзолата.

Упътвания:

1. Създайте конзолно C# приложение с име "Nums1To20".
2. Напишете 20 команди `Console.WriteLine()`, една след друга, за да отпечатате числата от 1 до 20.

2. Бонус точки

Дадено е **цяло число** – брой точки. Върху него се начисляват **бонус точки** по правилата, описани по-долу. Да се напише програма, която пресмята **бонус точките** за това число и **общия брой точки** с бонусите.

- Ако числото е **до 100** включително, бонус точките са **5**.
- Ако числото е **по-голямо от 100**, бонус точките са **20%** от числото.
- Ако числото е по-голямо от **1000**, бонус точките са **10%** от числото.
- Допълнителни бонус точки (начисляват се отделно от предходните):
 - За **четно** число >> + 1 т.
 - За число, което **завършва на 5** >> + 2 т.

Подсказка:

- Основните бонус точки можете да изчислите с **if-else-if-else-if** конструкция (имате 3 случая).
- Допълнителните бонус точки можете да изчислите с **if-else-if** конструкция (имате още 2 случая).

3. Всички латински букви

Напишете програма, която отпечатва всички букви от латинската азбука: **a, b, c, ..., z**.

Подсказка: можете да завъртите **for**-цикъл от **"a"**; до **"z"**; (освен числа може да въртите в цикъл и букви).

4. Ромбче от звездички

Напишете програма, която чете цяло положително число **n**, въведено от потребителя, и печата **ромбче от звездички** с размер **n** като в примерите по-долу:

вход	изход	вход	изход	вход	изход	вход	изход
1	*	2	* * * *	3	* * * * * * * * *	4	* * * * * * * * * * * * * * * *

Подсказки:

❑ Разделете ромба на горна и долна част и ги печатайте с два отделни цикъла.

❑ За **горната част** завъртете цикъл за **row** от **1** до **n**:

o Отпечатайте **n-row** интервала.

o Отпечатайте **"*"**.

o Отпечатайте **row-1** пъти **"*"**.

❑ **Долната част** отпечатайте аналогично на горната с цикъл от **1** до **n-1**.

5. Пирамида от числа

Напишете програма, която чете цяло число n , въведено от потребителя, и отпечатва **пирамида от числа** като в примерите:

вход	изход
7	1 2 3 4 5 6 7

вход	изход
10	1 2 3 4 5 6 7 8 9 10

вход	изход
12	1 2 3 4 5 6 7 8 9 10 11 12

вход	изход
15	1 2 3 4 5 6 7 8 9 10 11 12 13 14 15

Подсказка:

☐ С **два вложени цикъла** печатайте пирамида от числа: на първия ред едно число, на втория ред 2 числа, на третия ред 3 числа и т.н.

☐ В отделен **брояч** пазете колко числа сте отпечатали до момента (и кое е текущото число). Когато стигнете n , излезте внимателно от двата вложени цикъла с **break** или **return**.

6. Reverse an Array of Integers

Напишете програма, която чете масив от цели числа и извежда прочетения масив на обратно. Входът се състои от число n (броя елементи) + n цели числа, всеки от които е на отделен ред. Отпечатайте изхода на един ред (числата да бъдат разделени по празно място).

