

5. Аналогично на решението на зад. 18.

7. Намираме кои са цифрите на числото както в зад. 5 и зад. 18. За всяка от тях проверяваме дали се дели на 7. Тази проверка ни връща 0 или 1. Полученият резултат на практика е броят на делящите се на 7 цифри в числото. Остава само да проверим дали този брой е четен.

8. Знаем всички прости числа, които са по-малки от 23, т.е. всички възможности за $x = 2, 3, 5, 7, 11, 13, 17, 19$. Използвайте факта, че булевите изрази $x \leq 2, x \leq 3, \dots, x \leq 19$ връщат резултат 0 или 1. Какво намираме, когато съберем резултатите от тези булеви изрази?

11. Има грешка в условието. Трябва да бъде: „... Декларирайте втора променлива `integer_variable` и я инициализирайте с `double_variable...`“. Задачата използва само теорията за преобразуване на типове променливи, която можете да намерите в теория.pdf на стр. 5. Съответните стойности са разделени със „->“ на същия принцип, на който са разделени стойностите в 1 зад. със „:“.

12. Хубаво е да се запомни: преобразуване на `char symbol`, който е равен на цифра, в `int`, равен на същата цифра (ред 3):

```
char symbol = '2';  
cout << symbol; //изкарва 2, но това 2 е всъщност от тип char; има аски код 50; вж. таблицата  
int symbol_to_number = symbol - '0';  
cout << symbol_to_number; // изкарва 2, но вече не е char, а int;
```

След като имате преобразувани символите в цифри, единственото, което остава, е да ги превърнете в трицифрено число – `first_symbol_to_number*100 + second_symbol_to_number*10 + third_symbol_to_number`.

13. Разгледайте примера:

`x = 123456`, нека търсената цифра е `digit`

```
- ако k е 1, digit = x % 10;  
- ако k е 2, digit = x / 10 % 10;  
- ако k е 3, digit = x / 100 % 10;  
- ако k е 4, digit = x / 1000 % 10;  
- ако k е 5, digit = x / 10000 % 10;  
...
```

Каква зависимост забелязвате между `x`, `k` и `digit`?

14. Простичък "Hello World!", но с вашето име. ☺

16. Има няколко начина, от които може да си изберете само един – закръгляте надолу десетичното число с `floor` или преобразувате към цяло число пак по няколко начина (теорията, стр. 5). Аналогично на зад. 2е) в първия случай и на зад. 11 във втория случай.

17. Аналогично на зад. 2щ).

19. Забележете, че XOR на два елемента е 0 само ако двата елемента са равни или и двата елемента са 0. Освен това `x XOR 0 == x`. Какво ни дава това? Обяснение може да намерите на: <https://www.youtube.com/watch?v=2Bn5uBnmptU> и на <https://www.geeksforgeeks.org/find-the-number-occurring-odd-number-of-times/>.

22. Вж. <https://www.w3resource.com/c-programming-exercises/for-loop/c-for-loop-exercises-41.php>