

СУ „Св. Климент Охридски“, ФМИ

СПЕЦИАЛНОСТ „СОФТУЕРНО ИНЖЕНЕРСТВО“

Увод в програмирането, 2020-2021 г.**Задачи за домашно № 1**

1. Митко има нужда да си купи лаптоп. За целта, той събира пари, като на всеки свой нечетен рожден ден получава играчка, която после успява да продаде за М лева. Също така на всеки свой четен рожден ден, той получава пари в брой – на втория си рожден ден, получава 30 лева, а на всеки следващ четен рожден ден, тази сума се увеличава с по 30 лева. Освен това, на всеки рожден ден, на който получава пари в брой, на Митко му се налага да похарчи 5 лева, за да почерпи приятелите си. Напишете програма, която да провери дали на N-тия си рожден ден, Митко е събрал достатъчно пари да си купи лаптоп за Х лева. Програмата да въвежда от клавиатурата N, X и M, в посочения ред и да извежда на конзолата „yes“ и сумата, която е в повече ако Митко има достатъчно пари или „no“ и сумата, която не достига – в противен случай.

Пример:

Вход	Изход
12 500 10	yes 160
20 2000 30	no 100

2. Да се напише програма, която въвежда цели положителни числа от клавиатурата. Въвеждането на числата приключва когато се въведе нула или отрицателно число. Програмата да изведе на екрана второто по-големина от въведените числа. **Ако са въведени по-малко от две различни числа, да се изведе подходящ код за грешка (например -1).**

Пример:

Вход	Изход
3	25
25	
4	
12	
9	
7	
10	
56	
8	
-1	

3. Да се напише програма, която въвежда от клавиатурата 3 цифри и образува и отпечатва в конзолата най-малкото **число-палиндром¹**, което може да се образува от въведените цифри.

Пример:

Вход	Изход
8	17871
1	
7	
1	111
1	
1	
5	30503
0	
3	

¹ Палиндром се нарича такава последователност от символи, която се чете по един и същ начин, както от ляво надясно, така и от дясно наляво.

4. Да се напише програма, която приема като вход от клавиатурата цяло положително число $N \in [1, 10^9]$, и извежда в конзолата три цели положителни числа m , p и q , такива че $m \cdot p \cdot q = N$ и сумата им е минимална.

Пример:

Вход	Изход
72	3 4 6
51	1 3 17

5. Да се състави програма, която приема от клавиатурата цяло число $N \in [2, 10^9]$ и извежда на екрана броят на цифрите след десетичната запетая в записа на числото, ако $1/N$ има краен десетичен запис и NO - в противен случай.

Пример:

Вход	Изход
7	NO
8	3
16000	7

Пояснения:

1. Всички задачи носят по 2 точки.
2. Всички задачи ще бъдат проверени автоматично за преписване. Файловете с голямо съвпадение ще бъдат проверени ръчно от лектора и при установено плагиатство ще бъдат анулирани.
3. Опитайте се да напишете максимално ефективен код, както по отношение на брой редове, така и по отношение на време за изпълнение. Помислете къде може да се намали броят на повторенията на циклите или да се намали броят на променливите, които използвате за решаване на задачата.
4. За решаване на задачите не се допуска използване на масиви и STL функции.
5. Предадените от вас решения трябва да са написани на езика C/C++ и да могат да се компилират успешно на Visual C++ или GCC.
6. Всяка задача от домашното трябва да бъде решена в точно един, отделен файл. Името на файла трябва да бъде в следния формат:

fnXXXXX_d1_N_CC.cpp, където:

- XXXXX е вашият факултетен номер
 - N е номерът на задачата
 - CC указва кой компилатор сте използвали. Стойността му може да бъде "gcc" за GCC или "vc" за Visual C++.
7. Архивирайте всички файлове, които предавате в един архивен файл, компресиран в стандартен zip формат, със следното име:

UP_20-21_fnXXXXX_d1.zip, където XXXXX е вашият факултетен номер

8. Файловете с решенията, които предавате трябва да са оформени съгласно добрите практики за оформяне на кода, за които се говори по време на лекции и упражнения. Ще се отнемат точки за неинформативни имена на променливи, неизползване на подходящи константи и т.н.
9. Всички предадени програми трябва следят за некоректно въведени входни данни от потребителя, в зависимост от условието на задачата.
10. Файловете с решенията може да съдържат само стандартните символи с кодове от 0-127 (не се разрешава използване на кирилица, например в стринговете или коментарите!).
11. Първото нещо във всеки от файловете, които предавате, трябва да бъде коментарен блок, който носи информация за съдържанието на файла. Този коментар трябва да изглежда точно така, както е показано по-долу, като в него попълните информация за Вас. За улеснение, просто копирайте дадения по-долу блок и попълнете в него необходимите данни, вместо текста, маркиран с ъглови скоби. Обърнете внимание, че на първия ред след наклонената черта има две звезди и че във файловете не може да се съдържат символи на кирилица.

```
/**
 *
 * Solution to homework assignment 1
 * Introduction to programming course
 * Faculty of Mathematics and Informatics of Sofia University
 * Winter semester 2020/2021
 *
 * @author <вашето име>
 * @idnumber <вашият факултетен номер>
 * @task <номер на задача>
 * @compiler <използван компилатор - GCC или VC>
 *
 */
```

Например един попълнен блок за студент с име Иван Иванов, ф.н. 12345, който предава задача 2, компилирана с GCC, трябва да изглежда така:

```
/**
 *
 * Solution to homework assignment 1
 * Introduction to programming course
 * Faculty of Mathematics and Informatics of Sofia University
 * Winter semester 2020/2021
 *
 * @author Ivan Ivanov
 * @idnumber 12345
 * @task 2
 * @compiler GCC
 *
 */
```

12. Предадени домашни, които не отговарят на условията от точки 4-11 ще бъдат оценени с 0 точки.