

СУ „Св. Климент Охридски“, ФМИ

СПЕЦИАЛНОСТ „СОФТУЕРНО ИНЖЕНЕРСТВО“

Увод в програмирането, 2022-2023 г.

Задачи за домашно № 4

1. Да се напише програма на езика C++, която прочита от клавиатурата цяло, положително число $N \in [3, 50]$, последвано от N на брой символни низа. Да се отпечата в конзолата всички от тях, които представят валидно double число, сортирани във възходящ ред (по големина на числата). За отбелязване на дробните числа да се приема както символът '.' (точка), така и символът ',' (запетая). В случай, че няма въведени валидни double числа, да се изведе съобщение "no valid numbers".

Пояснения:

- Постарайте се да минимизирате използваната памет когато запазвате валидните double числа. В този смисъл не е удачно да запазвате дробно число в символен низ.
- Ако желаете да показвате числата с по-голяма точност след десетичната запетая, може да използвате методът `setprecision()` (<https://en.cppreference.com/w/cpp/io/manip/setprecision>)

Пример:

Вход	Изход
4 22.34 Adgfasdg 21.3 45	21.3 22.34 45

2. Да се напише програма на езика C++, която последователно прочита от клавиатурата два символни низа S и W (всеки с не повече от 255 символа) и отпечата на екрана началните индекси на всички поднизове на S , които са анаграми на W . В случай, че няма в S няма анаграми на W , да се изведе съобщение "no anagrams".

Примери:

Вход	Изход
isopenpoerty ope	2, 6
sdjsds ds	0, 3, 4
fmirulzz alfa	"no anagrams"

3. Даден е низ, който съдържа отварящи "(" и затварящи ")" скоби. Ще го наричаме „затворен“, ако след всяка отваряща скоба следва затваряща скоба някъде в низа и „правилен“, ако всеки затворен под-низ има равен брой отварящи и затварящи скоби.

Да се напише програма, която прочита от клавиатурата символен низ с не повече от 255 символа и извежда в конзолата "correct" ако той е затворен и правилен и "incorrect" - в противен случай. В случай, че низът е с повече от 255 символа, да се изведе съобщение "wrong input".

Примери:

Вход	Изход
ab(oisjdoi399)lk((lklk))()l	correct
(so i (j(o)i))	correct
(siio(lk)((9)	incorrect

4. Дадени са два легена с вместимост съответно n и m литра, където n и m са цели числа и $m, n \in [1, 20]$. Да се напише програма която намира **най-малкият брой** стъпки, необходими за да се получат точно k литра вода в единият от съдовете. Валидни са следните стъпки:

- Напълване на леген с вода
- Изливане на водата от леген
- Пресипване на вода от един леген в друг. Приема се, че при тази стъпка, няма загуби на вода от разливане и пресипването завършва когато някой от легените е празен или пълен.

Вход на програмата са три числа, съответно n , m и k , а изход е броят стъпки или -1 ако е невъзможно да се получат k литра. Ползвайте стандартния вход и изход.

Примери:

Последна промяна: 09.03.2023 г.; 13:00

Вход	Изход
5 3 2	2
8 5 6	6
2 3 4	-1

Допълнителни инструкции за решаване и предаване на домашното:

1. Всички коректно решени задачи носят по 2,5 точки.
2. Обърнете внимание, че в условието на всички задачи е посочено множеството от допустимите стойности на входните данни.
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. Всички задачи ще бъдат проверени автоматично за преписване. Файловете с голямо съвпадение ще бъдат проверени ръчно от преподавателите и при установено плагиатство ще бъдат анулирани.
5. **Всички решения на задачите от домашното трябва да бъдат предадени преди събота, 13 януари 2023, 23:59 ч.**
6. За решаване на задачите **не се допуска използване на STL функции.**
7. **Предадените от вас решения трябва да са написани на езика C/C++ и да могат да се компилират успешно на GCC.**
8. **Всяка задача от домашното трябва да бъде решена в точно един, отделен файл.** Името на файла трябва да бъде в следния формат:

fnXXXXX_d4_N.cpp, където:

- XXXXX е вашият факултетен номер
 - N е номерът на задачата.
9. Предаването на домашното се извършва в Github, според линка за предаване посочен в Moodle, като задачите се commit-ват в главната директория на repository-то. В Actions може да видите дали предадени задачи се компилират и дали примерните входове от условието се изпълняват успешно. След края на срока за предаване задачите ще бъдат тествани допълнително и с други тестове.
 10. 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.
 11. Файловете с решенията може да съдържат само стандартните ASCII символи с кодове от 0-127 (не се разрешава използване на кирилица, например в стринговете или коментарите!).
 12. Всички предадени програми трябва следят за некоректно въведени входни данни от потребителя, в зависимост от условието на задачата.
 13. В началото на всеки един от файловете, които предавате, трябва има коментарен блок, който носи информация за съдържанието на файла. Този коментарен блок 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 4
 * Introduction to programming course
 * Faculty of Mathematics and Informatics of Sofia University
 * Winter semester 2022/2023
 *
 * @author <вашето име>
 * @idnumber <вашият факултетен номер>
 * @task <номер на задача>
 *
 */
```

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

```
/**
 *
 * Solution to homework assignment 4
 * Introduction to programming course
 * Faculty of Mathematics and Informatics of Sofia University
 * Winter semester 2022/2023
 *
 * @author Ivan Ivanov
 * @idnumber 12345
 * @task 2
 *
 */
```

14. Предадени домашни, които не отговарят на условията от точки 5-12 от инструкциите описани по-горе ще бъдат оценени с 0 точки.