

СУ „Св. Климент Охридски“, ФМИ

СПЕЦИАЛНОСТ „СОФТУЕРНО ИНЖЕНЕРСТВО“

Увод в програмирането, 2020-2021 г.

Задачи за домашно за поправителната сесия

1. Да се напише програма на C++, която въвежда от стандартния вход цяло число Y в интервала $[0, 2100]$ и цяло число M в интервала $[0, 1000]$. Програмата да извежда на стандартния изход следващите след Y, M на брой високосни години.

Подберете сами подходящи входни данни, за да тествате Вашето решение на задачата.

2. Да се напише на C++ функция $F()$, която за дадена редица от цели положителни числа проверява дали в нея се съдържат Числа на Фибоначи и да ги извежда на стандартния изход сортирани. Да не се извеждат повтарящи се числа. Да се напише функция $main()$, която изпълнява $F()$ върху 3 локално дефинирани масива.

Пример:

Вход	Изход
3 11 5 1 7 12	1 3 5
3 0 0 1 12 8	0 1 3 8

3. Да се напише програма на C++, която за въведен от потребителя от стандартния вход символен низ, намира *максималната дума* и я извежда на стандартния изход.

Пояснения:

- **Дума** се нарича всяка последователност от символи, която не включва интервал и/или табулация ('\\t'). Броят на интервалите и табулациите не е ограничен.
- **Максимална дума в низа** е тази дума, която има най-голяма сума от ASCII кодовете на символите, които я съставят.

Примери:

Вход	Изход
This is a string.	string.
zzzzZzz AAAAAAAAAAAAAA	zzzzZzz

4. Да се напише функция F() на C++, която по зададени като параметри два символни низа и цяло число N, низа вмъква втория низ в първия на позиция N (т.е. наподобява операцията paste). Да се напише функция main(), която изпълнява F() върху въведени от потребителя два низа и число N. Резултатът от изпълнението на функцията main(), да се изведе в конзолата. За решаване на задачата не може да се използват string и/или vector, както и библиотеки и/или външни функции, т.е. такива, които не са написани от вас.

Подберете сами подходящи входни данни, за да тествате Вашето решение на задачата.

Пояснения:

1. Задача 1 носи 2 точки, задачи 2 и 3 носят по 2,5 точки, а задача 4 - 3 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. Счита се, че при въвеждането на данните, потребителят ще спазва типа им. Ако не е упоменато друго, всички програми да извеждат в конзолата -1 ако входните данни, подадени от потребителя са извън дефиниционното множество на задачата и да прекратяват изпълнението си.
5. **За решаване на задача 4 не се допуска използването на vector и/или string.**
6. Предадените от вас решения трябва да могат да се компилират успешно на Visual C++ или GCC.
7. Всяка задача от домашното трябва да бъде решена в точно един, отделен файл. Името на файла трябва да бъде в следния формат:

fnXXXXX_dS_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++.
8. Архивирайте всички файлове, които предавате в един архивен файл, компресиран в стандартен zip формат, със следното име:

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

9. 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.
10. Всички предадени програми трябва следят за некоректно въведени входни данни от потребителя, в зависимост от условието на задачата.
11. 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!).
12. 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 S
 * 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 S
 * 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
 *
 */
```

13. Предадени домашни, които не отговарят на условията от точки 5-12 ще бъдат оценени с 0 точки.