

Оформяне 2

Програмиране за 10 клас

28 юни 2017г. (вариант Г)

Време: 80 мин

Важно: Задачите, които предавате, трябва да се компилират! Некомпилиращи се файлове получават 0 точки.

Задача 1

Да се дефинира структурата **book_t**, която съдържа в себе си име (до 50 символа), година на издаване и брой продадени копия. Да се дефинира функцията **float estimate_rating(struct book_t)**, която казва ориентировъчен рейтинг на книгата, като го пресмята по формулата **(годината * копията) / дължината на името**.

Да се дефинира функцията **int rating_between(struct book_t[10], int min, int max)**, която намира и връща броя книги с рейтинг между **min** и **max**.

Задача 2

Дефинирайте структурата **book_t** от горната задача. Да се дефинира структурата **bookstore_t**, която съдържа масив до 10 книги и текущия брой книги в книжарницата.

Да се дефинира функцията **int sorted_add(struct bookstore_t*, struct book_t)**, която добавя книгата към книжарницата, ако тя не е пълна. Приемаме, че книгите са сортирани по години във възходящ ред и нашата книга трябва да се добави на първата възможна позиция, така че да не нарушава реда на годините. Функцията да връща индекса на който е добавена книгата, а ако няма място -1.

Задача 3

От конзолата се прочита последователност от символи (1 по 1 или накуп) до EOF. Да се дефинира функцията **int numeric_sequence_count(char*)**, която връща броя числа в текста, като едно число може да е съставено от цифрите от 0 до 9 или точка.

Например:

```
str = "12 3.14 sdf 2.563 12,34"  
numeric_sequence_count(str) // 5
```

Задача 4

Дефинирайте структурата **interval_t**, която има лява и дясна граница - цели числа.

Дефинирайте структурата **intervals_t**, която държи динамичен масив от интервали и бройката им. Всеки интервал в масива отговаря на намерен интервал в текста. Даден интервал седи между [и] скоби и е разделен с , т.е. [3,40] е интервал от 3 до 40..

Направете четене на текст до достигане до EOF.

Дефинирайте функцията **struct intervals_t get_all_intervals(char*)**, която намира всички интервали в текста и ги връща.

Например:

Изречението "The quick[10,16] brown fox[3,6] jumps over[3,10] the lazy[5,5] dog" ще има 4 интервала - [10 - 16], [3 - 6], [3 - 10] и [5 - 5].