

Вторая задача должна быть сдана до 13.03.2019 включительно. В случае невыполнения данного требования студенту дается дополнительное задание.

1. Текстовый файл содержит строки длиной не более 512 символов, состоящие из слов, разделенных пробелами и знаками табуляции. Написать функцию, которая из всех слов файла, состоящих только из букв английского алфавита и начинающихся на заглавную букву, находит самое короткое слово.

Имя файла и полученное слово должны передаваться через параметры функции. Функция должна возвращать 0, если файл удалось открыть, и -1, если файл не удалось открыть. Функция должна иметь описание:

```
int search(const char *SInputFile, char *word);
```

2. Текстовый файл содержит строки длиной не более 512 символов, состоящие из слов, разделенных пробелами и знаками табуляции. Написать функцию, которая из всех слов файла, являющихся текстовым представлением целых чисел, находит слово, являющееся текстовым представлением наименьшего целого числа.

Имя файла и полученное слово должны передаваться через параметры функции. Функция должна возвращать 0, если файл удалось открыть, и -1, если файл не удалось открыть. Функция должна иметь описание:

```
int search(const char *SInputFile, char *word);
```

3. Текстовый файл содержит строки длиной не более 512 символов, состоящие из слов, разделенных пробелами и знаками табуляции. Написать функцию, которая из всех слов файла, состоящих только из букв английского алфавита и содержащих заглавную букву, находит самое длинное слово.

Имя файла и полученное слово должны передаваться через параметры функции. Функция должна возвращать 0, если файл удалось открыть, и -1, если файл не удалось открыть. Функция должна иметь описание:

```
int search(const char *SInputFile, char *word);
```

4. Текстовый файл содержит строки длиной не более 512 символов, состоящие из слов, разделенных пробелами, точками и запятыми. Написать функцию, которая из всех слов файла, состоящих только из заглавных букв английского алфавита и начинающихся на букву *A*, находит самое короткое слово.

Имя файла и полученное слово должны передаваться через параметры функции. Функция должна возвращать 0, если файл удалось открыть, и -1, если файл не удалось открыть. Функция должна иметь описание:

```
int search(const char *SInputFile, char *word);
```

5. Текстовый файл содержит строки длиной не более 512 символов, состоящие из слов, разделенных пробелами, точками и запятыми. Написать функцию, которая из всех слов файла, являющихся текстовым представлением целых чисел и начинающихся на *0*, находит слово, являющееся текстовым представлением наименьшего целого числа.

Имя файла и полученное слово должны передаваться через параметры функции. Функция должна возвращать 0, если файл удалось открыть, и -1, если файл не удалось открыть. Функция должна иметь описание:

```
int search(const char *SInputFile, char *word);
```

6. Текстовый файл содержит строки длиной не более 512 символов, состоящие из слов, разделенных пробелами, точками и запятыми. Написать функцию, которая из всех слов файла, состоящих только из заглавных букв английского алфавита и заканчивающихся на букву *A*, находит самое длинное слово.

Имя файла и полученное слово должны передаваться через параметры функции. Функция должна возвращать 0, если файл удалось открыть, и -1, если файл не удалось открыть. Функция должна иметь описание:

```
int search(const char *SInputFile, char *word);
```

7. Текстовый файл содержит строки длиной не более 512 символов, состоящие из слов, заключенных в двойные кавычки. Информация вне слов должна игнорироваться. Считается, что в каждой строке находится четное количество кавычек.

Написать функцию, которая из всех слов файла, состоящих только из букв английского алфавита и начинающихся на заглавную букву, находит самое короткое слово.

Имя файла и полученное слово должны передаваться через параметры функции. Функция должна возвращать 0, если файл удалось открыть, и -1, если файл не удалось открыть. Функция должна иметь описание:

```
int search(const char *SInputFile, char *word);
```

8. Текстовый файл содержит строки длиной не более 512 символов, состоящие из слов, заключенных в двойные кавычки. Информация вне слов должна игнорироваться. Считается, что в каждой строке находится четное количество кавычек.

Написать функцию, которая из всех слов файла, являющихся текстовым представлением целых чисел, находит слово, являющееся текстовым представлением наименьшего целого числа.

Имя файла и полученное слово должны передаваться через параметры функции. Функция должна возвращать 0, если файл удалось открыть, и -1, если файл не удалось открыть. Функция должна иметь описание:

```
int search(const char *SInputFile, char *word);
```

9. Текстовый файл содержит строки длиной не более 512 символов, состоящие из слов, заключенных в двойные кавычки. Информация вне слов должна игнорироваться. Считается, что в каждой строке находится четное количество кавычек.

Написать функцию, которая из всех слов файла, состоящих только из букв английского алфавита и содержащих заглавную букву, находит самое длинное слово.

Имя файла и полученное слово должны передаваться через параметры функции. Функция должна возвращать 0, если файл удалось открыть, и -1, если файл не удалось открыть. Функция должна иметь описание:

```
int search(const char *SInputFile, char *word);
```

10. Текстовый файл содержит строки длиной не более 512 символов, состоящие из слов, заключенных в двойные кавычки. Информация вне слов должна игнорироваться. Считается, что в каждой строке находится четное количество кавычек.

Написать функцию, которая из всех слов файла, состоящих только из заглавных букв английского алфавита и начинающихся на букву *A*, находит самое короткое слово.

Имя файла и полученное слово должны передаваться через параметры функции. Функция должна возвращать 0, если файл удалось открыть, и -1, если файл не удалось открыть. Функция должна иметь описание:

```
int search(const char *SInputFile, char *word);
```

11. Текстовый файл содержит строки длиной не более 512 символов, состоящие из слов, заключенных в двойные кавычки. Информация вне слов должна игнорироваться. Считается, что в каждой строке находится четное количество кавычек.

Написать функцию, которая из всех слов файла, являющихся текстовым представлением целых чисел и начинающихся на 0, находит слово, являющееся текстовым представлением наименьшего целого числа.

Имя файла и полученное слово должны передаваться через параметры функции. Функция должна возвращать 0, если файл удалось открыть, и -1, если файл не удалось открыть. Функция должна иметь описание:

```
int search(const char *SInputFile, char *word);
```

12. Текстовый файл содержит строки длиной не более 512 символов, состоящие из слов, заключенных в двойные кавычки. Информация вне слов должна игнорироваться. Считается, что в каждой строке находится четное количество кавычек.

Написать функцию, которая из всех слов файла, состоящих только из заглавных букв английского алфавита и заканчивающихся на букву *A*, находит самое длинное слово.

Имя файла и полученное слово должны передаваться через параметры функции. Функция должна возвращать 0, если файл удалось открыть, и -1, если файл не удалось открыть. Функция должна иметь описание:

```
int search(const char *SInputFile, char *word);
```

14. Текстовый файл содержит строки длиной не более 512 символов, состоящие из слов, разделенных пробелами и знаками табуляции. Написать функцию, которая из всех слов файла, состоящих только из букв английского алфавита и начинающихся на заглавную букву, находит самое короткое слово.

Имя файла и полученное слово должны передаваться через параметры функции. Функция должна возвращать 0, если файл удалось открыть, и -1, если файл не удалось открыть. Функция должна иметь описание:

```
int search(const char *SInputFile, char *word);
```

15. Текстовый файл содержит строки длиной не более 512 символов, состоящие из слов, разделенных пробелами и знаками табуляции. Написать функцию, которая из всех слов файла, являющихся текстовым представлением целых чисел, находит слово, являющееся текстовым представлением наименьшего целого числа.

Имя файла и полученное слово должны передаваться через параметры функции. Функция должна возвращать 0, если файл удалось открыть, и -1, если файл не удалось открыть. Функция должна иметь описание:

```
int search(const char *SInputFile, char *word);
```

16. Текстовый файл содержит строки длиной не более 512 символов, состоящие из слов, разделенных пробелами и знаками табуляции. Написать функцию, которая из всех слов файла, состоящих только из букв английского алфавита и содержащих заглавную букву, находит самое длинное слово.

Имя файла и полученное слово должны передаваться через параметры функции. Функция должна возвращать 0, если файл удалось открыть, и -1, если файл не удалось открыть. Функция должна иметь описание:

```
int search(const char *SInputFile, char *word);
```

17. Текстовый файл содержит строки длиной не более 512 символов, состоящие из слов, разделенных пробелами, точками и запятыми. Написать функцию, которая из всех слов файла, состоящих только из заглавных букв английского алфавита и начинающихся на букву *A*, находит самое короткое слово.

Имя файла и полученное слово должны передаваться через параметры функции. Функция должна возвращать 0, если файл удалось открыть, и -1, если файл не удалось открыть. Функция должна иметь описание:

```
int search(const char *SInputFile, char *word);
```

18. Текстовый файл содержит строки длиной не более 512 символов, состоящие из слов, разделенных пробелами, точками и запятыми. Написать функцию, которая из всех слов файла, являющихся текстовым представлением целых чисел и начинающихся на *0*, находит слово, являющееся текстовым представлением наименьшего целого числа.

Имя файла и полученное слово должны передаваться через параметры функции. Функция должна возвращать 0, если файл удалось открыть, и -1, если файл не удалось открыть. Функция должна иметь описание:

```
int search(const char *SInputFile, char *word);
```

19. Текстовый файл содержит строки длиной не более 512 символов, состоящие из слов, разделенных пробелами, точками и запятыми. Написать функцию, которая из всех слов файла, состоящих только из заглавных букв английского алфавита и заканчивающихся на букву *A*, находит самое длинное слово.

Имя файла и полученное слово должны передаваться через параметры функции. Функция должна возвращать 0, если файл удалось открыть, и -1, если файл не удалось открыть. Функция должна иметь описание:

```
int search(const char *SInputFile, char *word);
```

20. Текстовый файл содержит строки длиной не более 512 символов, состоящие из слов, заключенных в двойные кавычки. Информация вне слов должна игнорироваться. Считается, что в каждой строке находится четное количество кавычек.

Написать функцию, которая из всех слов файла, состоящих только из букв английского алфавита и начинающихся на заглавную букву, находит самое короткое слово.

Имя файла и полученное слово должны передаваться через параметры функции. Функция должна возвращать 0, если файл удалось открыть, и -1, если файл не удалось открыть. Функция должна иметь описание:

```
int search(const char *SInputFile, char *word);
```

21. Текстовый файл содержит строки длиной не более 512 символов, состоящие из слов, заключенных в двойные кавычки. Информация вне слов должна игнорироваться. Считается, что в каждой строке находится четное количество кавычек.

Написать функцию, которая из всех слов файла, являющихся текстовым представлением целых чисел, находит слово, являющееся текстовым представлением наименьшего целого числа.

Имя файла и полученное слово должны передаваться через параметры функции. Функция должна возвращать 0, если файл удалось открыть, и -1, если файл не удалось открыть. Функция должна иметь описание:

```
int search(const char *SInputFile, char *word);
```

22. Текстовый файл содержит строки длиной не более 512 символов, состоящие из слов, заключенных в двойные кавычки. Информация вне слов должна игнорироваться. Считается, что в каждой строке находится четное количество кавычек.

Написать функцию, которая из всех слов файла, состоящих только из букв английского алфавита и содержащих заглавную букву, находит самое длинное слово.

Имя файла и полученное слово должны передаваться через параметры функции. Функция должна возвращать 0, если файл удалось открыть, и -1, если файл не удалось открыть. Функция должна иметь описание:

```
int search(const char *SInputFile, char *word);
```

23. Текстовый файл содержит строки длиной не более 512 символов, состоящие из слов, заключенных в двойные кавычки. Информация вне слов должна игнорироваться. Считается, что в каждой строке находится четное количество кавычек.

Написать функцию, которая из всех слов файла, состоящих только из заглавных букв английского алфавита и начинающихся на букву *A*, находит самое короткое слово.

Имя файла и полученное слово должны передаваться через параметры функции. Функция должна возвращать 0, если файл удалось открыть, и -1, если файл не удалось открыть. Функция должна иметь описание:

```
int search(const char *SInputFile, char *word);
```

24. Текстовый файл содержит строки длиной не более 512 символов, состоящие из слов, заключенных в двойные кавычки. Информация вне слов должна игнорироваться. Считается, что в каждой строке находится четное количество кавычек.

Написать функцию, которая из всех слов файла, являющихся текстовым представлением целых чисел и начинающихся на 0, находит слово, являющееся текстовым представлением наименьшего целого числа.

Имя файла и полученное слово должны передаваться через параметры функции. Функция должна возвращать 0, если файл удалось открыть, и -1, если файл не удалось открыть. Функция должна иметь описание:

```
int search(const char *SInputFile, char *word);
```

25. Текстовый файл содержит строки длиной не более 512 символов, состоящие из слов, заключенных в двойные кавычки. Информация вне слов должна игнорироваться. Считается, что в каждой строке находится четное количество кавычек.

Написать функцию, которая из всех слов файла, состоящих только из заглавных букв английского алфавита и заканчивающихся на букву *A*, находит самое длинное слово.

Имя файла и полученное слово должны передаваться через параметры функции. Функция должна возвращать 0, если файл удалось открыть, и -1, если файл не удалось открыть. Функция должна иметь описание:

```
int search(const char *SInputFile, char *word);
```