

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, являющихся текстовым представлением 10-ных 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);
```

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, состоящих только из букв английского алфавита и содержащих заглавную букву, находит последнее 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);
```

3. 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 А, находит первое 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);
```

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, являющихся текстовым представлением целых 10-ных чисел и начинающихся на 1, 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);
```

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, состоящих только из заглавных букв английского алфавита и заканчивающихся на букву A, находит первое 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);
```

6. Текстовый файл содержит строки длиной не более 512 символов, состоящие из слов, заключенных в двойные кавычки. Информация вне слов должна игнорироваться. Написать функцию, которая из всех слов файла, состоящих только из букв английского алфавита и начинающихся на заглавную букву, находит первое 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 или в некоторой строке находится нечетное количество кавычек. 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. Написать функцию, которая из всех слов файла, являющихся текстовым представлением 10-ных 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 или в некоторой строке находится нечетное количество кавычек. 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. Написать функцию, которая из всех слов файла, состоящих только из букв английского алфавита и содержащих заглавную букву, находит последнее 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 или в некоторой строке находится нечетное количество кавычек. 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. Написать функцию, которая из всех слов файла, состоящих только из заглавных букв английского алфавита и начинающихся на букву А, находит последнее 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 или в некоторой строке находится нечетное количество кавычек. 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. Написать функцию, которая из всех слов файла, являющихся текстовым представлением 10-ных целых чисел, кратных 10, 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 или в некоторой строке находится нечетное количество кавычек. Функция должна иметь описание:

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

11. 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 10-ных целых чисел, кратных 3, 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, разделенных пробелами и знаками табуляции. Написать функцию, которая из всех симметричных слов файла, состоящих только из букв английского алфавита, находит последнее 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);
```

13. Текстовый файл содержит строки длиной не более 512 символов, состоящие из слов, разделенных пробелами, точками и запятыми. Написать функцию, которая находит последнее слово, имеющее максимальное количество повторений одного символа в своем составе.

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, точками и запятыми. Написать функцию, которая находит последнее слово, имеющее максимальное количество разных символов.

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, точками и запятыми. Написать функцию, которая из всех слов файла, состоящих не только из заглавных букв английского алфавита, имеющих длину больше 7, 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);
```

16. Текстовый файл содержит строки длиной не более 512 символов, состоящие из слов, заключенных в двойные кавычки. Информация вне слов должна игнорироваться. Написать функцию, которая находит первое слово, которое повторяется (рядом) максимальное количество раз.

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 или в некоторой строке находится нечетное количество кавычек. Функция должна иметь описание:

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