

## Сдать решение задачи D0-Отсортировать слова

Ограничение времени: 1 с

Ограничение реального времени: 5 с

Ограничение памяти: 64M

### Отсортировать слова

Дана строка, состоящая из английских букв и пробелов. В конце строки символ точка. Все слова разделены одним пробелом. Необходимо составить из слов односвязный список и упорядочить по алфавиту. Список необходимо удалить в конце программы. Для сравнение строк можно использовать strcmp. Необходимо использовать данную структуры организации списка.

```
struct list {  
    char word[20];  
    struct list *next;  
}
```

Необходимо реализовать односвязный список и обслуживающие его функции.

```
add_to_list  
swap_elements  
print_list  
delete_list
```

### Примеры

#### Входные данные

```
efg abcd zzz.
```

#### Результат работы

```
abcd efg zzz
```

## Сдать решение задачи D1-Вид сверху

Ограничение времени: 1 с

Ограничение реального времени: 5 с

Ограничение памяти: 64M

### Вид сверху

В программе описана структура для хранения двоичного дерева:

```
typedef struct tree {  
    datatype key;  
    struct tree *left, *right;  
} tree;
```

Необходимо реализовать функцию, которая печатает вид дерева сверху. Функция должна строго соответствовать прототипу:

```
void btUpView(tree *root)
```

### Примеры

#### Входные данные

```
10 5 15 3 7 13 18 1 6 14 0
```

#### Результат работы

```
1 3 5 10 15 18
```

## Сдать решение задачи D2-Найти брата

Ограничение времени: 1 с

Ограничение реального времени: 5 с

Ограничение памяти: 64М

### Найти брата

В программе описано двоичное дерево:

```
typedef struct tree {  
    datatype key;  
    struct tree *left, *right;  
    struct tree *parent; //ссылка на родителя  
} tree;
```

Требуется реализовать функцию, которая по ключу возвращает адрес соседнего элемента - брата. Если такого ключа нет или у узла нет брата, то необходимо вернуть 0.  
Прототип функции:

```
tree * findBrother(tree *root, int key)
```

### Примеры

#### Входные данные

```
10 5 15 3 7 13 18 1 6 14 0 3
```

#### Результат работы

```
7
```