**Лабораторная работа**

**ИССЛЕДОВАНИЕ И ОЦЕНКА АЛГОРИТМОВ ПОИСКА**

**Цель работы**. Разработка программ, реализующих различные алгоритмы поиска, и оценка их временной и пространственной сложности.

**Задание.**

Разработать алгоритм и программу дихотомического поиска. В качестве исходных данных использовать массив целых чисел, который вводится с клавиатуры. Аргумент поиска – число.

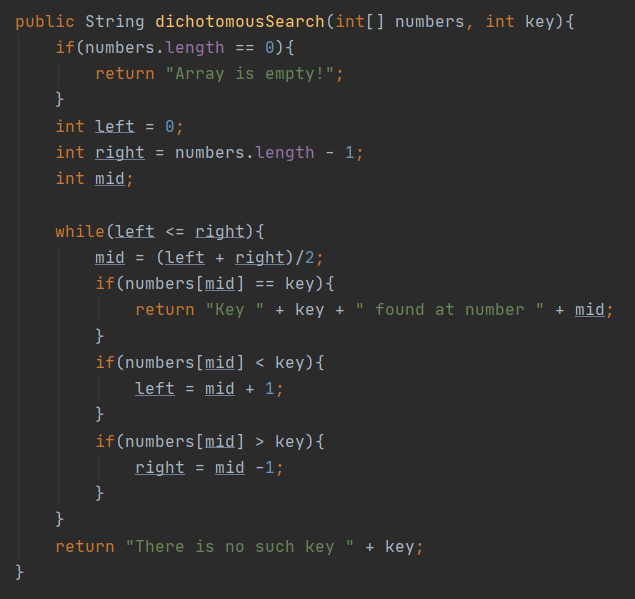
**Словесное описание заданного алгоритма поиска.**

 Классический [алгоритм](https://ru.wikipedia.org/wiki/%D0%90%D0%BB%D0%B3%D0%BE%D1%80%D0%B8%D1%82%D0%BC) поиска элемента в отсортированном массиве, использующий дробление массива на половины.

Поиск элемента в отсортированном массиве:

1. Определение значения элемента в середине структуры данных. Полученное значение сравнивается с ключом.
2. Если ключ меньше значения середины, то поиск осуществляется в первой половине элементов, иначе — во второй.
3. Поиск сводится к тому, что вновь определяется значение серединного элемента в выбранной половине и сравнивается с ключом.
4. Процесс продолжается до тех пор, пока не будет найден элемент со значением ключа или не станет пустым интервал для поиска.

**Код программы.**

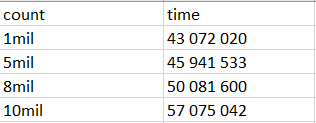


**Формулы верхней оценки временной и емкостной сложности заданного алгоритма.**

Верхняя оценка сложности алгоритма: О*(log(n)*)

Ёмкостная сложность алгоритма: О(1)

**Результаты экспериментальной оценки временной и емкостной сложности заданного алгоритма**.



int left – 4 bytes

int righr – 4 bytes

int mid – 4 bytes

4 + 4 + 4 = 12 bytes

O(1)