

# **”Quicksort” algorithm implementation**

## User guide

Krzysztof Kundera  
krzykun358@student.polsl.pl

FACULTY OF AUTOMATIC CONTROL, ELECTRONICS AND COMPUTER SCIENCE  
Computer Science Course

## 1. Project contents

My task was to create my own implementation of the quicksort algorithm. I chose Java and IntelliJ IDE, as I am familiar with Java, but unfamiliar with IntelliJ which seems popular among my friends.

Project contains a Sorter class that has a number of public functions:

1. Two constructors, one used in testing and the other suggested for User usage (because of simplicity)
2. A getter function for the DataArray - to lookup the results of the application work.
3. A trigger function – function that takes the data array as a parameter and launches application calculations.
4. Main function, for command line compatibility

Application is intended for command-line usage and is sorting only Integer numbers.

Main functionality – sorting – is covered with automatic test group that consists of six tests.

## 2. Usage

Every user can use the application through the command line. Call command takes just one argument which is a filename that contains an Integer number sequence. After every calculation, application will save it's work on the sequence to a file called "Output.txt". Please be sure to back- up any files made with the Application to avoid files being overwritten.

## 3. Application control flow and algorithm

Application always checks the command line arguments first. If they are present, they are analyzed, and saved. Then, a BufferedReader and FileReader classes are used to try and open a file. If that operation succeeds, input is converted from String type values into an array of Integer values. Main function then proceeds to call the quicksort algorithm (as described by the block diagram – see diagram folder in the repository).

Quicksort always checks, if the end conditions have been met. If not, number count is checked. Dependant on the count, Application returns the array (one element is always sorted), sorts two elements, or searches for a pivot in the sequence of numbers and then launches recursively the quicksort function twice, each for both ranges on each side of the pivot. Algorithm ends when we check every group counting two or a single value.

After the sorting is done, control switches back to the main function that calls the getter method, and then saves the results of the quicksort to the output file, called "Output.txt".

## 4. Possible development

First of all, algorithm can be generalized to handle different values than simple Integers, but due to time constraints I decided to keep this functionality as it is right now. Other area of possible development is the choice of pivot. Currently, this implementation takes the first number available from the start of the current numbers range. However, it has been proved that there are better methods, such as taking into consideration more values and random values.

## **5. Problems and feedback**

I warmly welcome any comments, thoughts or feedback. Please email me at [krzykun358@student.polsl.pl](mailto:krzykun358@student.polsl.pl) with any ideas or problems you might have, and I'll be more than happy to use them to make the application better!