

# One-Day Assignment 2 – algorithmic solution

Signs

# One-Day Assignment 2 – Pseudocode (code-like)

(not provided for this problem, as the complete pseudocode would be too similar to a working Java program)

# One-Day Assignment 2 – Pseudocode (English Description)

Store all input into an array of Strings called *arr*.

As a stable sort is required, the Java `sort()` method under the `Arrays` class can be used.

However, we provide the `sort()` method with a custom comparator that compares only the middle character(s) of each string, as calling `sort()` without a custom comparator would cause Java to sort by comparing the entire string.

This can be accomplished by having the comparator take the **substring of the middle character(s) from both strings**, and then using the resulting substrings to perform standard Java String comparison.

Once done, iterate through *arr* and print out the strings in *arr*, beginning from index 0.

# One-Day Assignment 2 – Finding the middle

We can use the *substring* method from Java's String API to get the middle character(s). Consider *str1* with length  $n$ .

- Case 1: Odd number of characters
  - Get the character in the middle with index  $\underline{n/2}$
- Case 2: Even number of characters
  - You need the two characters at index  $\underline{n/2 - 1}$  and  $\underline{n/2}$

Note that *substring(beginIndex, endIndex)* is inclusive for *beginIndex* but **exclusive** for *endIndex*.