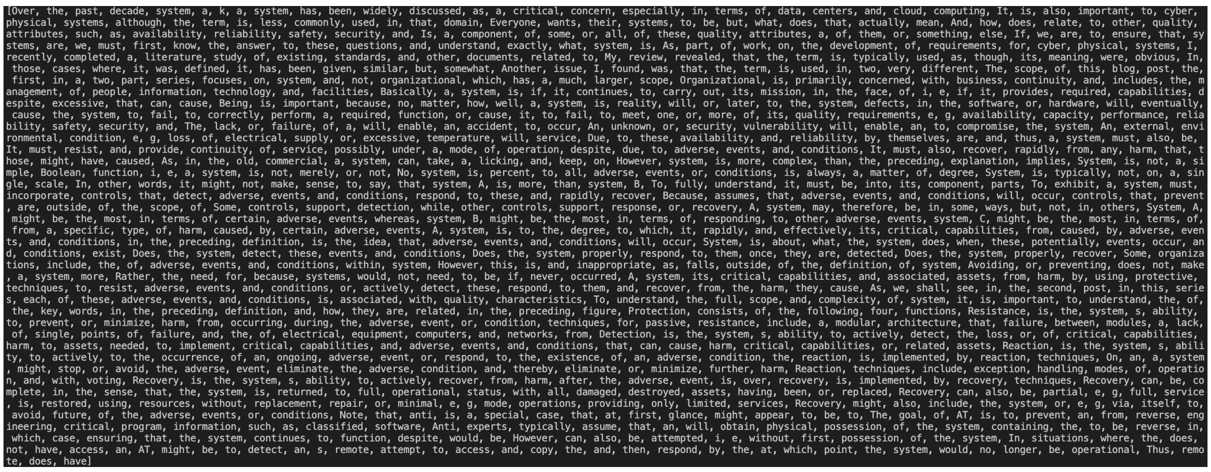
CMT219 – Part 1 Report

1. Functionality

A) Figure 1 is a screenshot of the result for running my code. It shows an ArrayList of strings as required, which contains the valid words in the given document based on the given vocabulary.

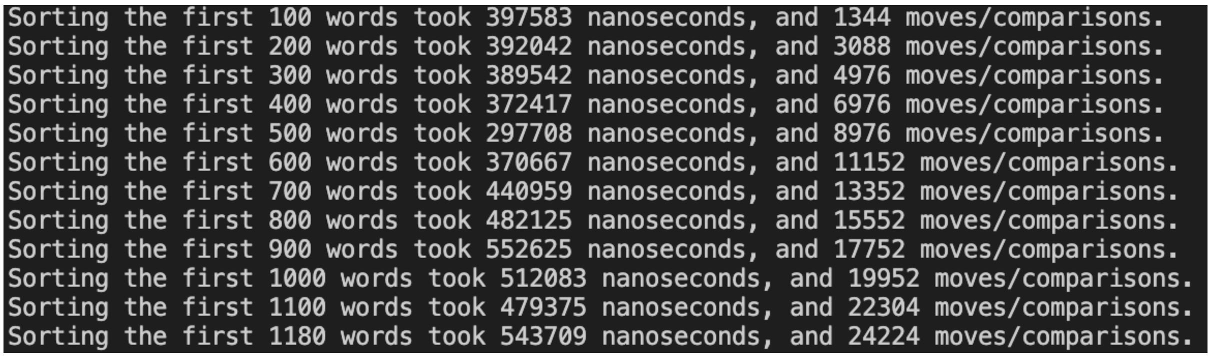


```
[Over, the, past, decade, system, a, k, a, system, has, been, widely, discussed, as, a, critical, concern, especially, in, terms, of, data, centers, and, cloud, computing, It, is, also, important, to, cyber,
physical, systems, although, the, term, is, less, commonly, used, in, that, domain, Everyone, wants, their, systems, to, be, but, what, does, that, actually, mean, And, how, does, relate, to, other, quality,
attributes, such, as, availability, reliability, safety, security, and, Is, a, component, of, some, or, all, of, these, quality, attributes, a, of, them, or, something, else, If, we, are, to, ensure, that, sy
stems, are, we, must, first, know, the, answer, to, these, questions, and, understand, exactly, what, system, is, As, part, of, work, on, the, development, of, requirements, for, cyber, physical, systems, I,
recently, completed, a, literature, study, of, existing, standards, and, other, documents, related, to, My, review, revealed, that, the, term, is, typically, used, as, though, its, meaning, were, obvious, In,
 those, cases, where, it, was, defined, it, has, been, given, similar, but, somewhat, Another, issue, I, found, was, that, the, term, is, used, in, two, very, different, The, scope, of, this, blog, post, the,
 first, in, a, two, part, series, focuses, on, system, and, not, organizational, which, has, a, much, larger, scope, Organizational, is, primarily, concerned, with, business, continuity, and, includes, the, m
anagement, of, people, information, technology, and, facilities, Basically, a, system, is, if, it, continues, to, carry, out, its, mission, in, the, face, of, i, e, if, it, provides, required, capabilities, d
espite, excessive, that, can, cause, Being, is, important, because, no, matter, how, well, a, system, is, reality, will, or, later, to, the, system, defects, in, the, software, or, hardware, will, eventually,
 cause, the, system, to, fail, to, correctly, perform, a, required, function, or, cause, it, to, fail, to, meet, one, or, more, of, its, quality, requirements, e, g, availability, capacity, performance, relia
bility, safety, security, and, The, lack, or, failure, of, a, will, enable, an, accident, to, occur, An, unknown, or, security, vulnerability, will, enable, an, to, compromise, the, system, An, external, envi
ronmental, condition, e, g, loss, of, electrical, supply, or, excessive, temperature, will, service, Due, to, these, availability, and, reliability, by, themselves, are, and, thus, a, system, must, also, be,
It, must, resist, and, provide, continuity, of, service, possibly, under, a, mode, of, operation, despite, due, to, adverse, events, and, conditions, It, must, also, recover, rapidly, from, any, harm, that, t
hose, might, have, caused, As, in, the, old, commercial, a, system, can, take, a, licking, and, keep, on, However, system, is, more, complex, than, the, preceding, explanation, implies, System, is, not, a, si
mple, Boolean, function, i, e, a, system, is, not, merely, or, not, No, system, is, percent, to, all, adverse, events, or, conditions, is, always, a, matter, of, degree, System, is, typically, not, on, a, sin
gle, scale, In, other, words, it, might, not, make, sense, to, say, that, system, A, is, more, than, system, B, To, fully, understand, it, must, be, into, its, component, parts, To, exhibit, a, system, must,
incorporate, controls, that, detect, adverse, events, and, conditions, respond, to, these, and, rapidly, recover, Because, assumes, that, adverse, events, and, conditions, will, occur, controls, that, prevent
, are, outside, of, the, scope, of, Some, controls, support, detection, while, other, controls, support, response, or, recovery, A, system, may, therefore, be, in, some, ways, but, not, in, others, System, A,
 might, be, the, most, in, terms, of, certain, adverse, events, whereas, system, B, might, be, the, most, in, terms, of, responding, to, other, adverse, events, system, C, might, be, the, most, in, terms, of,
 from, a, specific, type, of, harm, caused, by, certain, adverse, events, A, system, is, to, the, degree, to, which, it, rapidly, and, effectively, its, critical, capabilities, from, caused, by, adverse, even
ts, and, conditions, in, the, preceding, definition, is, the, idea, that, adverse, events, and, conditions, will, occur, System, is, about, what, the, system, does, when, these, potentially, events, occur, an
d, conditions, exist, Does, the, system, detect, these, events, and, conditions, Does, the, system, properly, respond, to, them, once, they, are, detected, Does, the, system, properly, recover, Some, organiza
tions, include, the, of, adverse, events, and, conditions, within, system, However, this, is, and, inappropriate, as, falls, outside, of, the, definition, of, system, Avoiding, or, preventing, does, not, make
, a, system, more, Rather, the, need, for, because, systems, would, not, need, to, be, if, never, occurred, A, system, its, critical, capabilities, and, associated, assets, from, harm, by, using, protective,
techniques, to, resist, adverse, events, and, conditions, or, actively, detect, these, respond, to, them, and, recover, from, the, harm, they, cause, As, we, shall, see, in, the, second, post, in, this, serie
s, each, of, these, adverse, events, and, conditions, is, associated, with, quality, characteristics, To, understand, the, full, scope, and, complexity, of, system, it, is, important, to, understand, the, of,
 the, key, words, in, the, preceding, definition, and, how, they, are, related, in, the, preceding, figure, Protection, consists, of, the, following, four, functions, Resistance, is, the, system, s, ability,
to, prevent, or, minimize, harm, from, occurring, during, the, adverse, event, or, condition, techniques, for, passive, resistance, include, a, modular, architecture, that, failure, between, modules, a, lack,
 of, single, points, of, failure, and, the, of, electrical, equipment, computers, and, networks, from, Detection, is, the, system, s, ability, to, actively, detect, the, loss, or, of, critical, capabilities,
harm, to, assets, needed, to, implement, critical, capabilities, and, adverse, events, and, conditions, that, can, cause, harm, critical, capabilities, or, related, assets, Reaction, is, the, system, s, abili
ty, to, actively, to, the, occurrence, of, an, ongoing, adverse, event, or, respond, to, the, existence, of, an, adverse, condition, the, reaction, is, implemented, by, reaction, techniques, On, an, a, system
, might, stop, or, avoid, the, adverse, event, eliminate, the, adverse, condition, and, thereby, eliminate, or, minimize, further, harm, Reaction, techniques, include, exception, handling, modes, of, operatio
n, and, with, voting, Recovery, is, the, system, s, ability, to, actively, recover, from, harm, after, the, adverse, event, is, over, recovery, is, implemented, by, recovery, techniques, Recovery, can, be, co
mplete, in, the, sense, that, the, system, is, returned, to, full, operational, status, with, all, damaged, destroyed, assets, having, been, or, replaced, Recovery, can, also, be, partial, e, g, full, service
, is, restored, using, resources, without, replacement, repair, or, minimal, e, g, mode, operations, providing, only, limited, services, Recovery, might, also, include, the, system, or, e, g, via, itself, to,
 avoid, future, of, the, adverse, events, or, conditions, Note, that, anti, is, a, special, case, that, at, first, glance, might, appear, to, be, to, The, goal, of, AT, is, to, prevent, an, from, reverse, eng
ineering, critical, program, information, such, as, classified, software, Anti, experts, typically, assume, that, an, will, obtain, physical, possession, of, the, system, containing, the, to, be, reverse, in,
 which, case, ensuring, that, the, system, continues, to, function, despite, would, be, However, can, also, be, attempted, i, e, without, first, possession, of, the, system, In, situations, where, the, does,
not, have, access, an, AT, might, be, to, detect, an, s, remote, attempt, to, access, and, copy, the, and, then, respond, by, the, at, which, point, the, system, would, no, longer, be, operational, Thus, remo
te, does, have]
```

Figure 1. Printed ArrayList of valid words in the document based on the vocabulary.

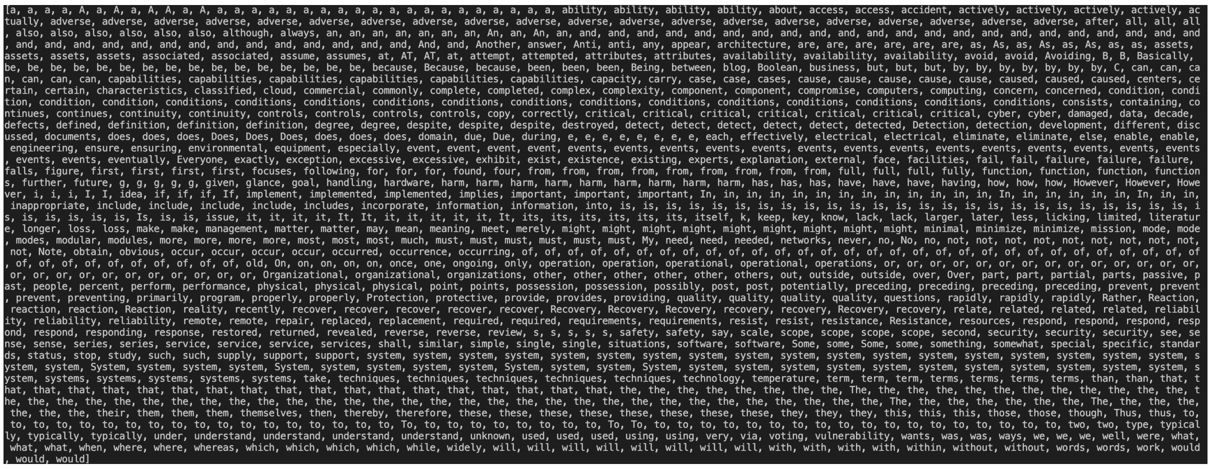
B) Figure 2 shows the results of calling implemented merge sort algorithm to sort the first 100, 200, … words in the valid word ArrayList. For each sorting, the time used and count of moves/comparisons during sorting is printed.

Figure 3 show the sorted ArrayList after the last sorting, i.e., soring all the elements in the ArrayList in alphabetical order, confirming the correctness of the implemented merge sort algorithm. The results contain multiple same words since ArrayList can contain elements with the same values.



```
Sorting the first 100 words took 397583 nanoseconds, and 1344 moves/comparisons.
Sorting the first 200 words took 392042 nanoseconds, and 3088 moves/comparisons.
Sorting the first 300 words took 389542 nanoseconds, and 4976 moves/comparisons.
Sorting the first 400 words took 372417 nanoseconds, and 6976 moves/comparisons.
Sorting the first 500 words took 297708 nanoseconds, and 8976 moves/comparisons.
Sorting the first 600 words took 370667 nanoseconds, and 11152 moves/comparisons.
Sorting the first 700 words took 440959 nanoseconds, and 13352 moves/comparisons.
Sorting the first 800 words took 482125 nanoseconds, and 15552 moves/comparisons.
Sorting the first 900 words took 552625 nanoseconds, and 17752 moves/comparisons.
Sorting the first 1000 words took 512083 nanoseconds, and 19952 moves/comparisons.
Sorting the first 1100 words took 479375 nanoseconds, and 22304 moves/comparisons.
Sorting the first 1180 words took 543709 nanoseconds, and 24224 moves/comparisons.
```

Figure 2. Calculated time used and counts of moves/comparisons during sorting.



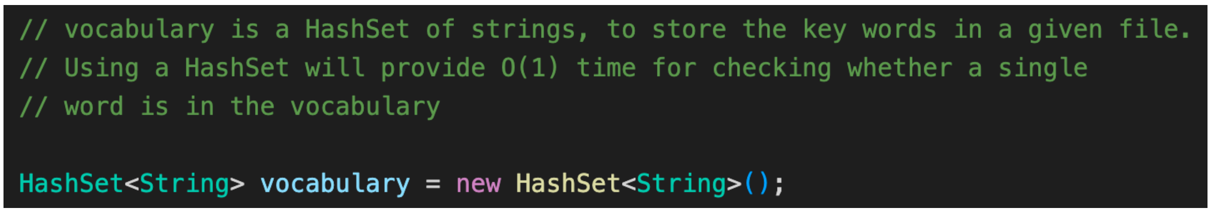
```
[a, a, a, a, A, a, A, a, A, A, a, A, a, a, a, a, a, a, a, a, a, a, a, a, a, a, a, a, a, a, a, a, ability, ability, ability, ability, about, access, access, accident, actively, actively, actively, actively, ac
tually, adverse, adverse, adverse, adverse, adverse, adverse, adverse, adverse, adverse, adverse, adverse, adverse, adverse, adverse, adverse, adverse, adverse, adverse, adverse, adverse, after, all, all, all
, also, also, also, also, also, also, although, always, an, an, an, an, an, an, an, An, an, An, an, and, and, and, and, and, and, and, and, and, and, and, and, and, and, and, and, and, and, and, and, and, and
, and, and, and, and, and, and, and, and, and, and, and, and, and, and, And, and, Another, answer, Anti, anti, any, appear, architecture, are, are, are, are, are, are, as, As, as, As, as, As, as, as, assets,
assets, assets, assets, associated, associated, assume, assumes, at, AT, AT, at, attempt, attempted, attributes, attributes, availability, availability, availability, avoid, avoid, Avoiding, B, B, Basically,
be, be, be, be, be, be, be, be, be, be, be, be, be, be, be, be, because, Because, because, been, been, been, Being, between, blog, Boolean, business, but, but, but, by, by, by, by, by, by, by, C, can, can, ca
n, can, can, can, capabilities, capabilities, capabilities, capabilities, capabilities, capabilities, capacity, carry, case, case, cases, cause, cause, cause, cause, cause, caused, caused, caused, centers, ce
rtain, certain, characteristics, classified, cloud, commercial, commonly, complete, completed, complex, complexity, component, component, compromise, computers, computing, concern, concerned, condition, condi
tion, condition, condition, conditions, conditions, conditions, conditions, conditions, conditions, conditions, conditions, conditions, conditions, conditions, conditions, conditions, consists, containing, co
ntinues, continues, continuity, continuity, controls, controls, controls, controls, copy, correctly, critical, critical, critical, critical, critical, critical, critical, cyber, cyber, damaged, data, decade,
defects, defined, definition, definition, definition, degree, degree, despite, despite, despite, destroyed, detect, detect, detect, detect, detect, detected, Detection, detection, development, different, disc
ussed, documents, does, does, does, Does, Does, Does, does, does, does, domain, due, Due, during, e, e, e, e, e, e, e, e, each, effectively, electrical, electrical, eliminate, eliminate, else, enable, enable,
 engineering, ensure, ensuring, environmental, equipment, especially, event, event, event, event, events, events, events, events, events, events, events, events, events, events, events, events, events, events
, events, events, eventually, Everyone, exactly, exception, excessive, excessive, exhibit, exist, existence, existing, experts, explanation, external, face, facilities, fail, fail, failure, failure, failure,
falls, figure, first, first, first, first, focuses, following, for, for, for, found, four, from, from, from, from, from, from, from, from, from, full, full, full, fully, function, function, function, function
s, further, future, g, g, g, g, g, given, glance, goal, handling, hardware, harm, harm, harm, harm, harm, harm, harm, harm, harm, has, has, has, have, have, have, having, how, how, how, However, However, Howe
ver, i, i, i, I, I, idea, if, if, if, If, implement, implemented, implemented, implies, important, important, important, In, in, in, in, in, in, in, in, in, in, in, in, in, In, in, in, in, in, in, In, in, in,
 inappropriate, include, include, include, include, includes, incorporate, information, information, into, is, is, is, is, is, is, is, is, is, is, is, is, is, is, is, is, is, is, is, is, is, is, is, is, is, i
s, is, is, is, is, is, Is, is, is, issue, it, it, it, it, It, It, it, it, it, it, it, It, its, its, its, its, its, its, itself, k, keep, key, know, lack, lack, larger, later, less, licking, limited, literatur
e, longer, loss, loss, make, make, management, matter, matter, may, mean, meaning, meet, merely, might, might, might, might, might, might, might, might, might, minimal, minimize, minimize, mission, mode, mode
, modes, modular, modules, more, more, more, more, most, most, most, much, must, must, must, must, must, must, My, need, need, needed, networks, never, no, No, no, not, not, not, not, not, not, not, not, not,
 not, Note, obtain, obvious, occur, occur, occur, occur, occurred, occurrence, occurring, of, of, of, of, of, of, of, of, of, of, of, of, of, of, of, of, of, of, of, of, of, of, of, of, of, of, of, of, of, of
, of, of, of, of, of, of, of, of, of, of, old, On, on, on, on, on, once, one, ongoing, only, operation, operation, operational, operational, operations, or, or, or, or, or, or, or, or, or, or, or, or, or, or,
 or, or, or, or, or, or, or, or, or, or, or, Organizational, organizational, organizations, other, other, other, other, other, others, out, outside, outside, over, Over, part, part, partial, parts, passive, p
ast, people, percent, perform, performance, physical, physical, physical, point, points, possession, possession, possibly, post, post, potentially, preceding, preceding, preceding, preceding, prevent, prevent
, prevent, preventing, primarily, program, properly, properly, Protection, protective, provide, provides, providing, quality, quality, quality, quality, questions, rapidly, rapidly, rapidly, Rather, Reaction,
 reaction, reaction, Reaction, reality, recently, recover, recover, recover, recover, recover, Recovery, Recovery, Recovery, recovery, recovery, Recovery, recovery, relate, related, related, related, reliabil
ity, reliability, reliability, remote, remote, repair, replaced, replacement, required, required, requirements, requirements, resist, resist, resistance, Resistance, resources, respond, respond, respond, resp
ond, respond, responding, response, restored, returned, revealed, reverse, reverse, review, s, s, s, s, s, safety, safety, say, scale, scope, scope, scope, scope, second, security, security, security, see, se
nse, sense, series, series, service, service, service, services, shall, similar, simple, single, single, situations, software, software, Some, some, Some, some, something, somewhat, special, specific, standar
ds, status, stop, study, such, such, supply, support, support, system, system, system, system, system, system, system, system, system, system, system, system, system, system, system, system, system, system, s
ystem, system, System, system, system, system, System, system, system, system, system, System, system, system, System, system, system, system, system, system, system, system, system, system, system, system, s
ystem, systems, systems, systems, systems, systems, take, techniques, techniques, techniques, techniques, techniques, technology, temperature, term, term, term, terms, terms, terms, terms, than, than, that, t
hat, that, that, that, that, that, that, that, that, that, that, that, that, that, that, that, that, that, the, the, the, the, the, the, the, the, The, the, the, the, the, the, the, the, the, the, the, the, t
he, the, the, the, the, the, the, the, the, the, the, the, the, the, the, the, the, the, the, the, the, the, the, the, the, the, the, the, the, the, the, the, The, the, the, the, the, the, the, The, the, the, the,
 the, the, the, their, them, them, them, themselves, then, thereby, therefore, these, these, these, these, these, these, these, these, they, they, they, this, this, this, those, those, though, Thus, thus, to,
 to, to, to, to, to, to, to, to, to, to, to, to, to, to, to, to, to, To, to, to, to, to, to, to, to, to, To, To, to, to, to, to, to, to, to, to, to, to, to, to, to, to, to, to, to, to, two, two, type, typical
ly, typically, typically, under, understand, understand, understand, understand, unknown, used, used, used, using, using, very, via, voting, vulnerability, wants, was, was, ways, we, we, we, well, were, what,
 what, what, when, where, where, whereas, which, which, which, which, while, widely, will, will, will, will, will, will, will, will, with, with, with, with, within, without, without, words, words, work, would
, would, would]
```

Figure 3. Sorted ArrayList of valid words in alphabetical order.

2. Design

A) As mentioned in the requirement document, the "two text files are relatively big, you should consider how to make your program efficient". Therefore, I choose to use the **HashSet<String>** data structure to store the vocabulary words, as shown in Figure 4.

Using HashSet could avoid redundant words in the set, and it could also provide a constant O(1) time complexity for testing whether a word in the document is contained in the HashSet or not. Since the document is relatively big (many words), using HashSet to improve the speed of checking each word is significant for the algorithm efficiency of the program.

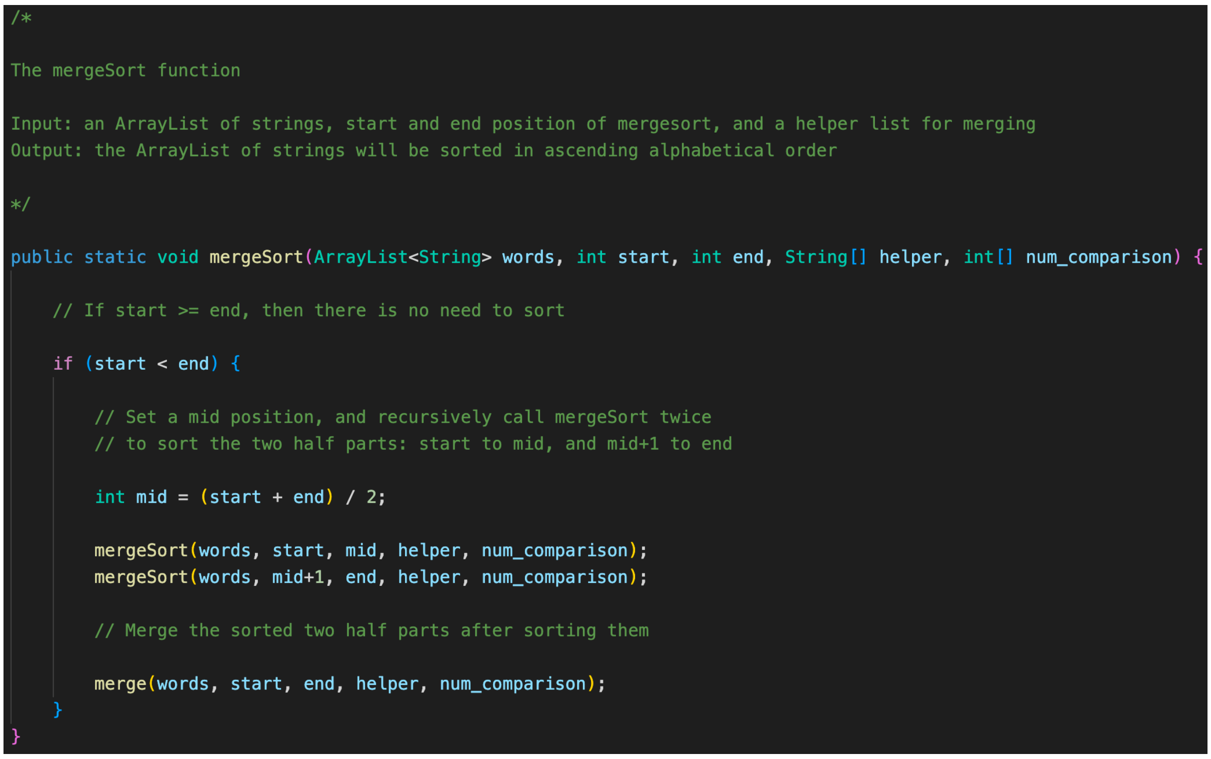


```
// vocabulary is a HashSet of strings, to store the key words in a given file.
// Using a HashSet will provide O(1) time for checking whether a single
// word is in the vocabulary

HashSet<String> vocabulary = new HashSet<String>();
```

Figure 4. Using HashSet data structure for the vocabulary.

B) For sorting, I implemented the merge sort algorithm as required, as shown in Figure 5. It partitioned the whole list into two half parts, and then recursively call merge sort algorithm to sort each of the two parts. Finally, a merge function is used to merge the sorted two half parts into one part. The termination condition of recursion is start >= end, i.e., whenever there is at most one element, which means there is no need to further partition or sort.

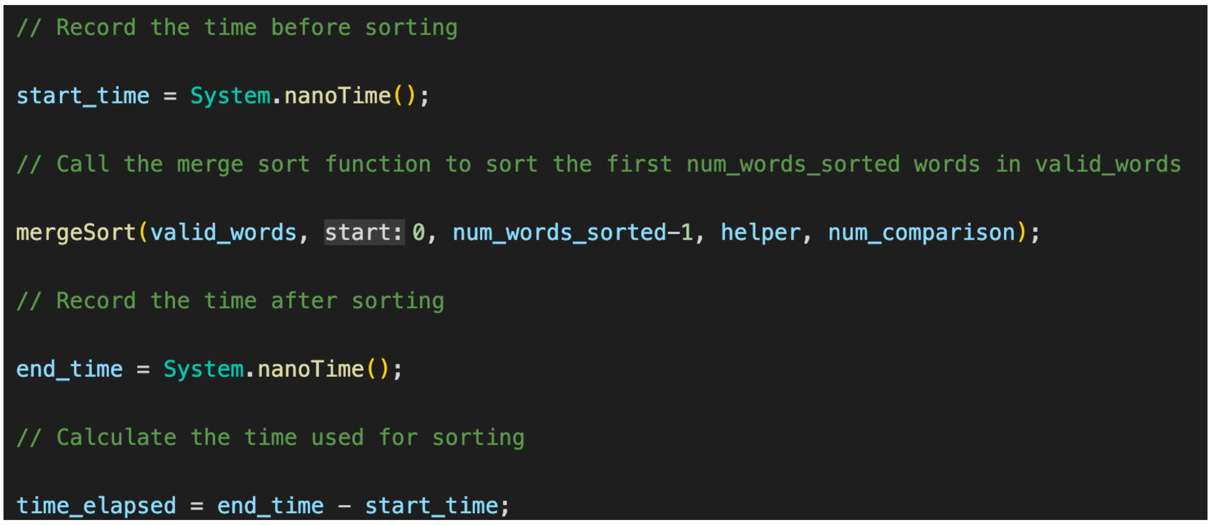
The argument of int[] num_compartion in the definition of mergeSort function is used to maintain the count of moves/comparisons during sorting. Using int[] data type here is because of Java passes by values for int type, which cannot be obtained after calling the functions.



```java
/*

The mergeSort function

Input: an ArrayList of strings, start and end position of mergesort, and a helper list for merging
Output: the ArrayList of strings will be sorted in ascending alphabetical order

*/

public static void mergeSort(ArrayList<String> words, int start, int end, String[] helper, int[] num_comparison) {

    // If start >= end, then there is no need to sort

    if (start < end) {

        // Set a mid position, and recursively call mergeSort twice
        // to sort the two half parts: start to mid, and mid+1 to end

        int mid = (start + end) / 2;

        mergeSort(words, start, mid, helper, num_comparison);
        mergeSort(words, mid+1, end, helper, num_comparison);

        // Merge the sorted two half parts after sorting them

        merge(words, start, end, helper, num_comparison);
    }
}
```

Figure 5. Implementation of merge sort.

To calculate the time used during sorting, I record the time before and after sorting, and calculate their difference, as shown in Figure 6.

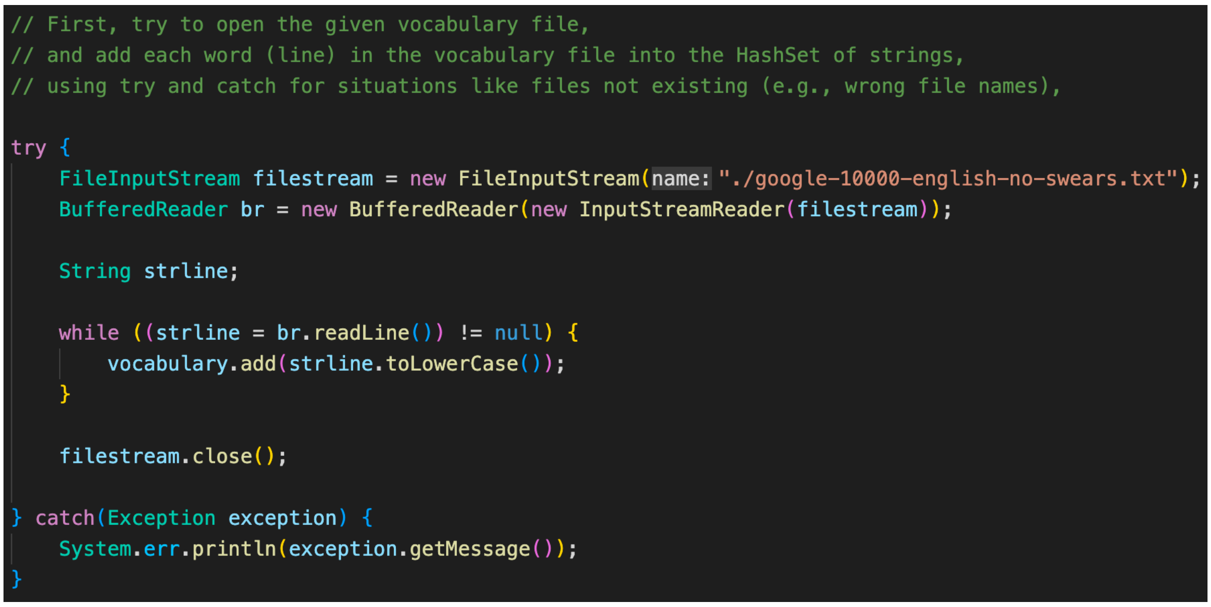


```java
// Record the time before sorting

start_time = System.nanoTime();

// Call the merge sort function to sort the first num_words_sorted words in valid_words

mergeSort(valid_words, start: 0, num_words_sorted-1, helper, num_comparison);

// Record the time after sorting

end_time = System.nanoTime();

// Calculate the time used for sorting

time_elapsed = end_time - start_time;
```

Figure 6. Calculate time used during sorting.

3. Ease of use and documentation

I use try and catch to make the code deal with invalid user input, such as file names which do not exist in the devices, as shown in Figure 7. In case of using non-existing file names, the program will not crash, and it will output a description of exception for fixing the issues, (e.g, ./google-10000-english-no-swears-wrong-filename.txt (No such file or directory)).



```java
// First, try to open the given vocabulary file,
// and add each word (line) in the vocabulary file into the HashSet of strings,
// using try and catch for situations like files not existing (e.g., wrong file names),

try {
    FileInputStream filestream = new FileInputStream(name: "./google-10000-english-no-swears.txt");
    BufferedReader br = new BufferedReader(new InputStreamReader(filestream));

    String strline;

    while ((strline = br.readLine()) != null) {
        vocabulary.add(strline.toLowerCase());
    }

    filestream.close();

} catch(Exception exception) {
    System.err.println(exception.getMessage());
}
```

Figure 7. Calculate time used during sorting.

As shown in above screenshots, the output is formatted in a easy to understand way, and the code is well commented with clear explanations.