**Homework #8**

**Functional programming and streams**

95 **points total**

All questions in this assignment refer to the 0 EnglishWords class in the hw8 package. This class uses the Java Stream API to process a file called words.txt. The words.txt file is supplied with the assignment, in the data directory. The words.txt file contains about 370,000 English words, each listed on a separate line.

All code written for this assignment must employ streaming operations from the java.util.stream package. Code that employs non-streaming operations will receive no credit.

The hw8 package also contains a test class which you can use to check your answers. Note that this package uses the **JUnit5** library, in contrast to most other packages in the course which use JUnit4.

Question 1. 5 5(5 points) Complete the countWords() method.

Question 2. 5 10(5 points) Complete the countWordsBeginningWith() method.

Question 3. 5 15(5 points) Complete the sumOfLengths() method.

Question 4. 5 20(5 points) Complete the sumOfLengthsContaining() method.

Question 5. 5 25(5 points) Complete the sumOfLengthsSquared() method.

Question 6. 10 35(10 points) Complete the countWordsOfLength() method.

Question 7. 10 45(10 points) Complete the printWordsLongerThan() method.

Question 8. 10 55(10 points) Complete the printWordsLongerThanParallel() method.

Question 9. 10 65(10 points) Complete the printWordsOfLenCapitalized() method.

Question 10. 15 80(15 points) Complete the joinAllContaining() method.

Question 11. 15 95(15 points) Complete the longest() method. As a fun challenge, try to do this using the reduce() operation. This question can also be solved using the max() operation, but we didn’t study this operation in class, and it will be better practice to use reduce().

Question 12. (Optional, ungraded) In class we covered only seven stream operations, but there are several others that are important and frequently used. These include min(), max(), average() (for the numeric stream types), limit(), and sorted(). Read about these operations and use them to implement some interesting new functionality for the EnglishWords class. For example, what is the average length of words in the word file? To use the average() method, you will need to read about the java.util.OptionalDouble class.

Question 13. (Optional, ungraded) Another very important use of streams is to aggregate frequency data via the collect() operation. This is a little more complex but well worth understanding. Study the method finalLetterFrequencies() in the EnglishWords class to see how it builds a Map storing the frequencies of final letters in the list of words. Implement another operation that computes frequencies. For example, build a Map containing the frequencies of certain common endings such as “-ed”, “-ing”, “-ious”.