Problem 1: Passing a Parameter via the Command Line (40%)

(a) Mapper output: (n, 2) (n, 3) (i, 2) (d, 10) (n, 3) (t, 3) (b, 4) (t, 4)
Reducer input: (n, [2, 3, 3]) (i, 2) (d, 10) (t, [3, 4]) (b, 4)
Reducer output: (n, 2.667) (i, 2.0) (d, 10.0) (t, 3.5) (b, 4.0)

(b) Are there any differences in the job execution?
Toolrunner is faster and more flexible and recompiling is required. All implementations of Tool need to implement Configurable. The run() method obtains the Configuration using Configurable's getConf() method.
By using Toolrunner, main() method does not invoke its own run() method directly. Instead, we call ToolRunner's static run() method, which takes care of creating a Configuration object for the Tool before calling its run() method. ToolRunner also uses a GenericOptionsParser to pick up any standard options specified on the command line and to set them on the Configuration instance.

(c) The job execution is similar as (b). The difference is that we use the Mapper's setup() method to retrieve the value of caseSensitive from command line.

(d) The average word lengths for the following letters:
- A: 3.89
- W: 4.46
- a: 3.08
- t: 3.73
- z: 4.67

(e) Command: hadoop jar awl.jar stubs.AvgWordLength -D caseSensitive=false shakespeare \ hw4_1e
The order matters, if we specify the parameter after the output directory, the input and output directory cannot be found.
The average word lengths for the following letters:
- a: 3.28
- w: 4.37
- z: 5.05

Problem 2: Word Count with a Partitioner (60%)

(a) Implement a Partitioner using the stubs SentimentPartitioner.java provided in your SVN repository.

(b) Observe the SentimentPartitionerTest.java program provided in your SVN repository. You can use it to test if your partitioner works correctly.

(c) Modify the WordCount Driver to use your Partitioner and the appropriate number of Reducers.
Positive words：405
Negative words：805
Neutral words：5215
Sentiment score s and the positivity score p:

$$s = \frac{positive - negative}{positive + negative} = \frac{405 - 805}{405 + 805} = \frac{-400}{1210} = -0.331$$

$$p = \frac{positive}{positive + negative} = \frac{405}{405 + 805} = \frac{405}{1210} = 0.335$$

Based on these statistics, I think Shakespeare's poems suggest negative emotions. Because s < 0 and p < 0.5, which implies that the number of positive words is less than the number of negative words.

(d) 1) The meaning of words are not always imply the sentiment of the sentences or paragraphs. For example, when we say "We are not happy", the emotion is negative but the positive word appears.
2) Sometimes positive words are more than negative words but positive words are largely concentrated on a few paragraphs. While the whole article's emotion can still be negative.

Improvement: we can split an article into several paragraphs and count the number of positive words and negative words separately. Then count the number of paragraphs with positive sentiment and negative sentiment to determine the overall sentiment. What is more, we can also search for words such as *no, not, never* around emotional words to identify its true meaning.