Cypher Algorithm:

Darren John Adams 14256232

Duncan Smallwood 13027205

Algorithm

1. Textfiles must be given.
    1. A .txt file with the encoded message.
    2. A .txt file with the dictionary words for 1,2 and 3 letters (Can be changed to use anything)
    3. A .txt file defining the punctuation of the language to ignore.

2. The algorithm starts by reading in all words into a list. This list is then purged of punctuations.

3. Next, the algorithm finds all duplicate words that are 1, 2 or 3 letters long.

4. Our algorithm assumes that these duplicate small words are trivial words and easy to decode. These duplicates are tested by shifting in increments of 1 up until 62. These new words are checked against our dictionary. If ALL of them match, we continue to perform this same shift on all of the words in the cipher text given.

5. Should our duplicates not yield any common shifting, we revert back to applying shifting again from (4) but using ALL 1, 2 and 3 letter words.

Speed

Our algorithm assumes a relatively small subset to work on first. Solving this is the quickest and easiest as it has the fewest elements.

If it has to use the entire set of 1,2 and 3 letter words, the computation to solve this would take longer.

Overall our program does not use efficient data structures, using ArrayLists, and as such is not as quick as it could be. We do, however, use logic to prevent an overload of computations needed by assuming on first pass.