Name : Parth Kalbhor

BE-AIML

Practical 1 Conflation algorithm

CODE:

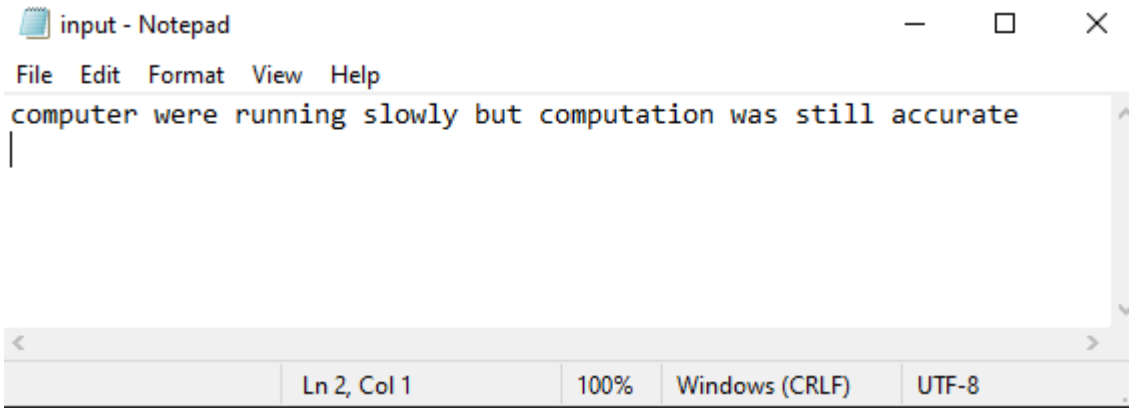
```
import java.io.*;
import java.util.*;
// --- Simplified Porter Stemmer implementation ---
class PorterStemmer {
   private char[] b;
   private int i, i_end;
   private static final int INC = 50;
   public PorterStemmer() {
      b = new char[INC];
      i = 0;
      i_end = 0;
   }
   public void add(char ch) {
      if (i == b.length) {
         char[] new_b = new char[i + INC];
         System.arraycopy(b, 0, new_b, 0, i);
         b = new_b;
      }
      b[i++] = ch;
   }
   public void reset() { i = 0; }
   public String toString() {
      return new String(b, 0, i_end);
   }
   public void stem() {
      String word = new String(b, 0, i);
      String result = stemWord(word);
      i_end = result.length();
```

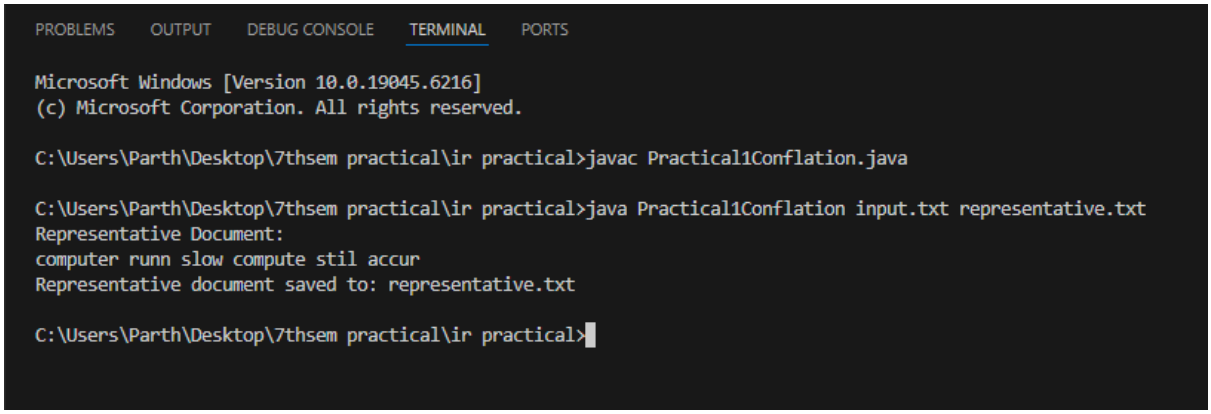
```
    b = result.toCharArray();
}
// --- Improved stemmer rules ---
private String stemWord(String word) {
    // plurals
    if (word.endsWith("sses")) return word.substring(0, word.length() - 2);
    if (word.endsWith("ies")) return word.substring(0, word.length() - 2);
    if (word.endsWith("s") && word.length() > 3) return word.substring(0, word.length() - 1);
    // past tense / continuous
    if (word.endsWith("ing") && word.length() > 4)
        return word.substring(0, word.length() - 3);
    if (word.endsWith("ed") && word.length() > 3)
        return word.substring(0, word.length() - 2);
    // adverbs
    if (word.endsWith("ly") && word.length() > 3)
        return word.substring(0, word.length() - 2);
    // noun → root
    if (word.endsWith("ation") && word.length() > 6)
        return word.substring(0, word.length() - 5) + "e";   // computation → compute
    if (word.endsWith("tion") && word.length() > 5)
        return word.substring(0, word.length() - 4) + "e";   // action → acte → act (rough)
    // adjectives
    if (word.endsWith("ness") && word.length() > 5)
        return word.substring(0, word.length() - 4);
    if (word.endsWith("ive") && word.length() > 4)
        return word.substring(0, word.length() - 3);
    if (word.endsWith("ate") && word.length() > 4)
        return word.substring(0, word.length() - 3);
    if (word.endsWith("ize") && word.length() > 4)
        return word.substring(0, word.length() - 3);
    if (word.endsWith("ous") && word.length() > 4)
        return word.substring(0, word.length() - 3);
    if (word.endsWith("ful") && word.length() > 4)
        return word.substring(0, word.length() - 3);
```

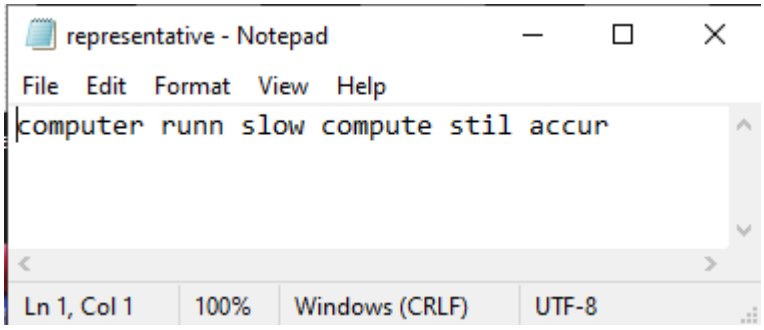
```
        // handle double consonants (runn → run, stopp → stop)
        if (word.length() > 3) {
            char last = word.charAt(word.length() - 1);
            char secondLast = word.charAt(word.length() - 2);
            if (last == secondLast) {
                return word.substring(0, word.length() - 1);
            }
        }
        return word;
    }
}
// --- Main Program ---
public class Practical1Conflation {
    public static void main(String[] args) throws Exception {
        if (args.length < 2) {
            System.out.println("Usage: java ConflationDemo <input.txt> <output.txt>");
            return;
        }
        String inputFile = args[0];
        String outputFile = args[1];
        // Expanded stopword list
        Set<String> stopwords = new HashSet<>(Arrays.asList(
            "the","is","are","a","an","and","of","in","to","it","on","for","with",
            "at","by","from","this","that","was","were","wa","but","be","been","being","am"
        ));
        // Read input file
        BufferedReader br = new BufferedReader(new FileReader(inputFile));
        StringBuilder sb = new StringBuilder();
        String line;
        while ((line = br.readLine()) != null) {
            sb.append(line.toLowerCase()).append(" ");
        }
        br.close();
        // Tokenize words
```

```
        String[] words = sb.toString().split("\\W+");
        // Stem + remove stopwords
        StringBuilder repDoc = new StringBuilder();
        for (String w : words) {
            if (w.isEmpty() || stopwords.contains(w)) continue;
            PorterStemmer stemmer = new PorterStemmer();
            for (char c : w.toCharArray()) stemmer.add(c);
            stemmer.stem();
            repDoc.append(stemmer.toString()).append(" ");
        }
        // Write representative document
        BufferedWriter bw = new BufferedWriter(new FileWriter(outputFile));
        bw.write(repDoc.toString().trim());
        bw.close();
        System.out.println("Representative Document:");
        System.out.println(repDoc.toString().trim());
        System.out.println("Representative document saved to: " + outputFile);
    }
}
```

OUTPUT :


input - Notepad
File Edit Format View Help
computer were running slowly but computation was still accurate
Ln 2, Col 1
100%
Windows (CRLF)
UTF-8


PROBLEMS
OUTPUT
DEBUG CONSOLE
TERMINAL
PORTS
Microsoft Windows [Version 10.0.19045.6216]
(c) Microsoft Corporation. All rights reserved.
C:\Users\Parth\Desktop\7thsem practical\ir practical>javac Practical1Conflation.java
C:\Users\Parth\Desktop\7thsem practical\ir practical>java Practical1Conflation input.txt representative.txt
Representative Document:
computer runn slow compute stil accur
Representative document saved to: representative.txt
C:\Users\Parth\Desktop\7thsem practical\ir practical>


representative - Notepad
File Edit Format View Help
computer runn slow compute stil accur
Ln 1, Col 1
100%
Windows (CRLF)
UTF-8