```
import java.io.BufferedReader;
import java.io.FileNotFoundException;
import java.io.FileReader;
import java.io.FileWriter;
import java.util.Random;

class node {
        char val;
        int pond =0;
        node bros;
        node son;
}//node2

class Analyser {

        node readingWindow (String chars, node theRoot){
                int i = 0;
                node pointer = theRoot;
                while (i < chars.length()){
                        if (pointer.son == null) {
                                node pointer2 = new node();
                                pointer.son = pointer2;
                        }//if
                        pointer = pointer.son;
                        while (pointer.bros != null && pointer.bros.val != chars.charAt
(i))
                                pointer = pointer.bros;
                        if (pointer.bros == null){
                                node pointer2 = new node();
                                pointer2.val = chars.charAt(i);
                                pointer.bros = pointer2;
                        }//if
                        pointer = pointer.bros;
                        pointer.pond++;
                        i++;
                }//while
                return theRoot;
        }//readingWindow
        node readingFile(String url, int windowSize, node theRoot) throws Exception {
                BufferedReader br = null;
            String line;
            try{
                br = new BufferedReader
                  (new FileReader(url));
              } catch(FileNotFoundException exc) {
                System.out.println("Erreur d'ouverture");
              }
            while ((line = br.readLine()) != null){
                for (int i=0; i<line.length()-windowSize; i++){
                        String tmp = line.substring(i, i+windowSize);
                        if (!tmp.contains(" ")){
                                System.out.println(tmp);
                                theRoot = readingWindow(tmp, theRoot);
                        }//if
                }//for
            }//while
            br.close();
            return theRoot;
        }//readingFile
        node run (int size) throws Exception{
            node theRoot = new node();
```

```
                theRoot = readingFile("/home/portable/workspace/SSI3/langevin/src/
res.txt", size, theRoot);
                return theRoot;
        }//run

}//analyser

public class Producer {
        Analyser a = new Analyser();
        String produceTokens (node theRoot){
                String toReturn ="";
                node pointer = theRoot;
                while (pointer.son != null){
                        pointer = pointer.son.bros;
                        int probSum = 0;
                        node ppointer = pointer;
                        while (ppointer != null){
                                probSum += ppointer.pond;
                                ppointer = ppointer.bros;
                        }//while
                        System.out.println("probSum = "+probSum);
                        int ind = new Random().nextInt(probSum+1);
                        while (pointer.bros != null && ind > 0){
                                ind -= pointer.pond;
                                pointer = pointer.bros;
                        }//while
                        toReturn += pointer.val;
                }//while pointer
                return toReturn;
        }//produceTokens
        String produceText (int nbOfWords, node theRoot){
                String toReturn = "";
                for (int i = 0; i<nbOfWords; i++){
                        toReturn = (toReturn + produceTokens(theRoot) + " ");
                        if (i%10 == 9) toReturn += " \n ";
                }//for
                return toReturn;
        }//produceText
        void run (int size, node theRoot) throws Exception {
                FileWriter fw = new FileWriter("/home/portable/workspace/SSI3/langevin/
src/test.txt");
                fw.write(produceText(size, theRoot));
                fw.flush();
                fw.close ();
        }//run
        public static void main (String [] args) throws Exception{
                Producer p = new Producer();
                node theRoot = p.a.run(4);
                p.run(500, theRoot);
        }//main
}//producer
```