NATIONAL INSTITUTE OF TECHNOLOGY SILCHAR

Cachar, Assam

B.Tech. VIIth Sem

Subject Code: CS-484

Subject Name: Cloud Computing

Submitted By:

Name	: Subhojit Ghimire
Sch. Id.	: 1912160
Branch	: CSE – B

Q. Write a MapReduce program to count k-mers (28-mers or 55-mers) of a DNA sequence.

➔ **Filename: KmerCount.java**

```
import java.io.IOException;
import java.util.StringTokenizer;

import org.apache.hadoop.conf.Configuration;
import org.apache.hadoop.fs.Path;
import org.apache.hadoop.io.*;
import org.apache.hadoop.mapreduce.*;
import org.apache.hadoop.mapreduce.lib.input.FileInputFormat;
import org.apache.hadoop.mapreduce.lib.output.FileOutputFormat;

public class KmerCount {

  public static class KmerMapper extends Mapper<LongWritable, Text, Text, IntWritable> {

    private final static IntWritable one = new IntWritable(1);
    private Text kmer = new Text();

    public void map(LongWritable key, Text value, Context context) throws IOException,
          InterruptedException {

      String line = value.toString().toUpperCase();
      int k = 28;
      //int k = 55; depending whether it is for 28 or 55
      // Loop over the line and extract k-mers
      for (int i = 0; i <= line.length() - k; i++) {
        String kmerStr = line.substring(i, i + k);
        kmer.set(kmerStr);
        context.write(kmer, one);
      }
    }
  }

  public static class KmerReducer extends Reducer<Text, IntWritable, Text, IntWritable> {

    private IntWritable result = new IntWritable();

    public void reduce(Text key, Iterable<IntWritable> values, Context context) throws
          IOException, InterruptedException {
      int sum = 0;
      for (IntWritable val : values) {
        sum += val.get();
      }
      result.set(sum);
      context.write(key, result);
    }
  }

  public static void main(String[] args) throws Exception {
    Configuration conf = new Configuration();
    Job job = Job.getInstance(conf, "kmer count");
    job.setJarByClass(KmerCount.class);
    job.setMapperClass(KmerMapper.class);
    job.setCombinerClass(KmerReducer.class);
    job.setReducerClass(KmerReducer.class);
    job.setOutputKeyClass(Text.class);
    job.setOutputValueClass(IntWritable.class);
    FileInputFormat.addInputPath(job, new Path(args[0]));
    FileOutputFormat.setOutputPath(job, new Path(args[1]));
    int k = Integer.parseInt(args[2]);
    job.getConfiguration().setInt("k", k);
    job.waitForCompletion(true);
  }
}
```

During executing, the format should be:
hadoop jar <jarAddress>.jar KmerCount <inputAddress> <outputAddress> <k-mers size arg>

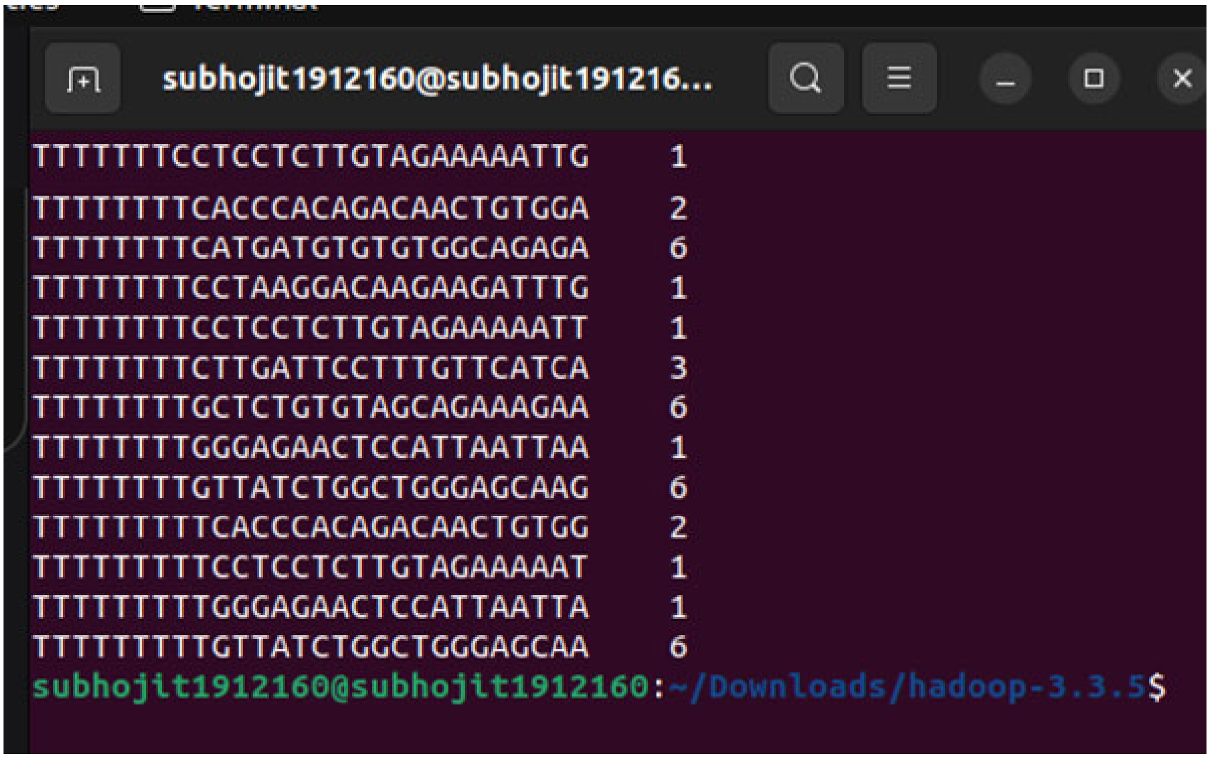
Foldername/Filename: /input/dnaSequence/human.txt
(DNA dataset: https://www.kaggle.com/datasets/nageshsingh/dna-sequence-dataset)

```
  sequence          class
  ATGCCCCAACTAAATACTACCGTATGGCCCACCATAATTACCCCCATACTCCTTACACTATTCCTCATC
       ACCCAACTAAAAATATTAAACACAAACTACCACCTACCTCCCTCACCAAAGCCCATAAAAATAAA
       AAATTATAACAAACCCTGAGAACCAAAATGAACGAAAATCTGTTCGCTTCATTCATTGCCCCCAC
       AATCCTAG…
  … (5,547,716 characters long)
```

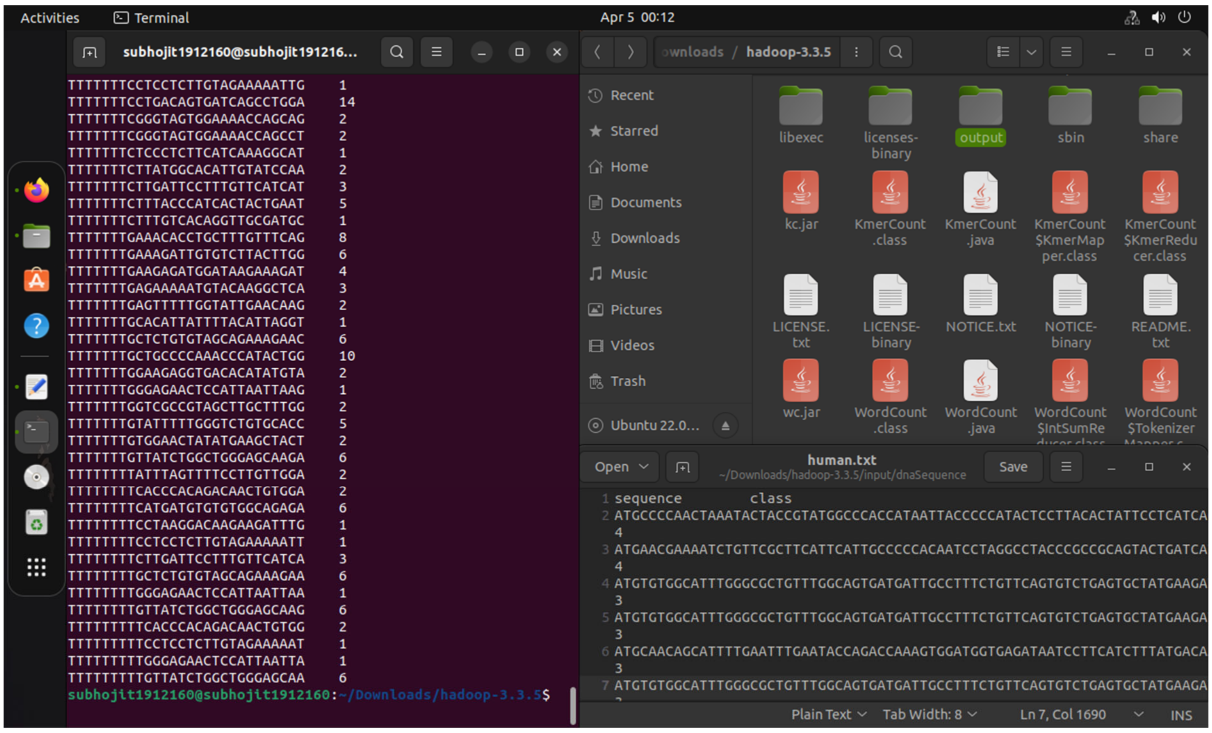
Execution:

```
  $ bin/hadoop com.sun.tools.javac.Main KmerCount.java
  $ jar cf kc.jar KmerCount*.class
  $ bin/hadoop jar kc.jar KmerCount input/dnaSequence output 28
  $ $ bin/hadoop fs -cat output/part-r-00000
```

Output:



```
TTTTTTTCCTCCTCTTGTAGAAAAATTG      1
TTTTTTTTCACCCACAGACAACTGTGGA      2
TTTTTTTTCATGATGTGTGTGGCAGAGA      6
TTTTTTTTCCTAAGGACAAGAAGATTTG      1
TTTTTTTTCCTCCTCTTGTAGAAAAATT      1
TTTTTTTTCTTGATTCCTTTGTTCATCA      3
TTTTTTTTGCTCTGTGTAGCAGAAAGAA      6
TTTTTTTTGGGAGAACTCCATTAATTAA      1
TTTTTTTTGTTATCTGGCTGGGAGCAAG      6
TTTTTTTTTCACCCACAGACAACTGTGG      2
TTTTTTTTTCCTCCTCTTGTAGAAAAAT      1
TTTTTTTTTGGGAGAACTCCATTAATTA      1
TTTTTTTTTGTTATCTGGCTGGGAGCAA      6
subhojit1912160@subhojit1912160:~/Downloads/hadoop-3.3.5$
```



```
TTTTTTTCCTCCTCTTGTAGAAAAATTG      1
TTTTTTTCCTGACAGTGATCAGCCTGGA      14
TTTTTTTCGGGTAGTGGAAAACCAGCAG      2
TTTTTTTCGGGTAGTGGAAAACCAGCCT      2
TTTTTTTCTCCCTCTTCATCAAAGGCAT      1
TTTTTTTCTTATGGCACATTGTATCCAA      2
TTTTTTTCTTGATTCCTTTGTTCATCAT      3
TTTTTTTCTTTACCCATCACTACTGAAT      5
TTTTTTTCTTTGTCACAGGTTGCGATGC      1
TTTTTTTGAAACACCTGCTTTGTTTCAG      8
TTTTTTTGAAAGATTGTGTCTTACTTGG      6
TTTTTTTGAAGAGATGGATAAGAAAGAT      4
TTTTTTTGAGAAAAATGTACAAGGCTCA      3
TTTTTTTGAGTTTTTGGTATTGAACAAG      2
TTTTTTTGCACATTATTTTACATTAGGT      1
TTTTTTTGCTCTGTGTAGCAGAAAGAAC      6
TTTTTTTGCTGCCCCAAACCCATACTGG      10
TTTTTTTGGAAGAGGTGACACATATGTA      2
TTTTTTTGGGAGAACTCCATTAATTAAG      1
TTTTTTTGGTCGCCGTAGCTTGCTTTGG      2
TTTTTTTGTATTTTTGGGTCTGTGCACC      5
TTTTTTTGTGGAACTATATGAAGCTACT      2
TTTTTTTGTTATCTGGCTGGGAGCAAGA      6
TTTTTTTTATTTAGTTTTCCTTGTTGGA      2
TTTTTTTTCACCCACAGACAACTGTGGA      2
TTTTTTTTCATGATGTGTGTGGCAGAGA      6
TTTTTTTTCCTAAGGACAAGAAGATTTG      1
TTTTTTTTCCTCCTCTTGTAGAAAAATT      1
TTTTTTTTCTTGATTCCTTTGTTCATCA      3
TTTTTTTTGCTCTGTGTAGCAGAAAGAA      6
TTTTTTTTGGGAGAACTCCATTAATTAA      1
TTTTTTTTGTTATCTGGCTGGGAGCAAG      6
TTTTTTTTTCACCCACAGACAACTGTGG      2
TTTTTTTTTCCTCCTCTTGTAGAAAAAT      1
TTTTTTTTTGGGAGAACTCCATTAATTA      1
TTTTTTTTTGTTATCTGGCTGGGAGCAA      6
subhojit1912160@subhojit1912160:~/Downloads/hadoop-3.3.5$
```

human.txt
~/Downloads/hadoop-3.3.5/input/dnaSequence

```
sequence        class
ATGCCCCAACTAAATACTACCGTATGGCCCACCATAATTACCCCCATACTCCTTACACTATTCCTCATCA
4
ATGAACGAAAATCTGTTCGCTTCATTCATTGCCCCCACAATCCTAGGCCTACCCGCCGCAGTACTGATCA
4
ATGTGTGGCATTTGGGCGCTGTTTGGCAGTGATGATTGCCTTTCTGTTCAGTGTCTGAGTGCTATGAAGA
3
ATGTGTGGCATTTGGGCGCTGTTTGGCAGTGATGATTGCCTTTCTGTTCAGTGTCTGAGTGCTATGAAGA
3
ATGCAACAGCATTTTGAATTTGAATACCAGACCAAAGTGGATGGTGAGATAATCCTTCATCTTTATGACA
3
ATGTGTGGCATTTGGGCGCTGTTTGGCAGTGATGATTGCCTTTCTGTTCAGTGTCTGAGTGCTATGAAGA
```