Lab 6

From the following link extract the weather data https://github.com/tomwhite/hadoop book/tree/master/input/ncdc/all Create a Map Reduce program to a) find average temperature for each year from NCDC data set. b) find the mean max temperature for every month.

Driver Code

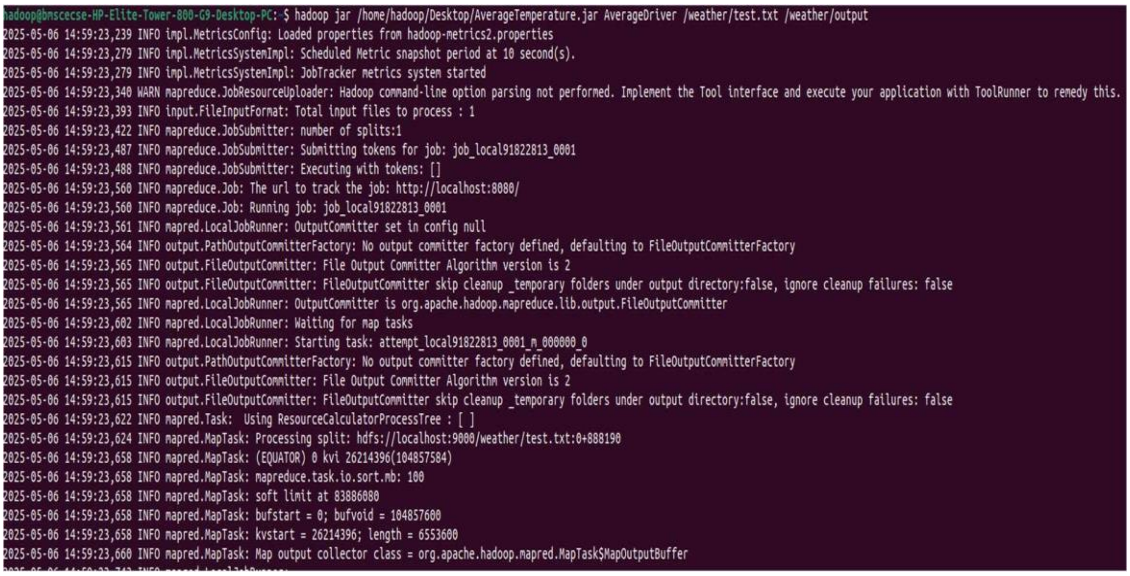
```
package temp;

import
org.apache.hadoop.conf.Configuration;
import org.apache.hadoop.fs.Path;

import org.apache.hadoop.io.IntWritable;
import org.apache.hadoop.io.Text;

import org.apache.hadoop.mapreduce.Job;

import
org.apache.hadoop.mapreduce.lib.input.FileInputForm
at; import
org.apache.hadoop.mapreduce.lib.output.FileOutputFo
rmat;

public class AverageDriver {

    public static void main(String[] args) throws Exception {

        if (args.length != 2) {
            System.err.println("Please enter both input and output parameters.");
            System.exit(-1);
        }

        // Creating a configuration and job instance
        Configuration conf = new Configuration();
        Job job = Job.getInstance(conf, "Average Calculation");

        job.setJarByClass(AverageDriver.class);

        // Input and output paths
        FileInputFormat.addInputPath(job, new Path(args[0]));
        FileOutputFormat.setOutputPath(job, new Path(args[1]));

        // Setting mapper and reducer classes
job.setMapperClass(AverageMapper.class);
        job.setReducerClass(AverageReducer.class);
        // Output key and value types
job.setOutputKeyClass(Text.class);
        job.setOutputValueClass(IntWritable.class);

        // Submitting the job and waiting for it to complete
        System.exit(job.waitForCompletion(true) ? 0 : 1);
    }
}
```

Mapper Code

```
package temp;

import java.io.IOException;

import
org.apache.hadoop.io.IntWritable;
import
org.apache.hadoop.io.LongWritable;
import org.apache.hadoop.io.Text;
import org.apache.hadoop.mapreduce.Mapper;

public class AverageMapper extends Mapper<LongWritable, Text, Text, IntWritable> {

    public static final int MISSING = 9999;

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

        String line = value.toString();

        // Extract year from fixed
position             String year =
line.substring(15, 19);          int
temperature;

        // Determine if there's a '+' sign         if
(line.charAt(87) == '+') {            temperature =
Integer.parseInt(line.substring(88, 92));
        } else {
            temperature = Integer.parseInt(line.substring(87, 92));
        }

        // Quality check character
        String quality = line.substring(92, 93);

        // Only emit if data is valid
        if (temperature != MISSING && quality.matches("[01459]")) {
            context.write(new Text(year), new IntWritable(temperature));
        }
    }
}
```

Reducer Code

```
package temp;

import java.io.IOException;

import
org.apache.hadoop.io.IntWritable;
import org.apache.hadoop.io.Text;
import org.apache.hadoop.mapreduce.Reducer;

public class AverageReducer extends Reducer<Text, IntWritable, Text, IntWritable> {

    @Override
    public void reduce(Text key, Iterable<IntWritable> values,
                        Context context) throws IOException, InterruptedException {

        int sumTemp = 0;
        int count = 0;

        for (IntWritable value : values) {
sumTemp += value.get();
            count++;
        }

        if (count > 0) {            int average =
sumTemp / count;
context.write(key, new
IntWritable(average));
        }
    }
}
```



```
hadoop@bmscecse-HP-Elite-Tower-800-G9-Desktop-PC:~$ hadoop jar /home/hadoop/Desktop/AverageTemperature.jar AverageDriver /weather/test.txt /weather/output
2025-05-06 14:59:23,239 INFO impl.MetricsConfig: Loaded properties from hadoop-metrics2.properties
2025-05-06 14:59:23,279 INFO impl.MetricsSystemImpl: Scheduled Metric snapshot period at 10 second(s).
2025-05-06 14:59:23,279 INFO impl.MetricsSystemImpl: JobTracker metrics system started
2025-05-06 14:59:23,340 WARN mapreduce.JobResourceUploader: Hadoop command-line option parsing not performed. Implement the Tool interface and execute your application with ToolRunner to remedy this.
2025-05-06 14:59:23,393 INFO input.FileInputFormat: Total input files to process : 1
2025-05-06 14:59:23,422 INFO mapreduce.JobSubmitter: number of splits:1
2025-05-06 14:59:23,487 INFO mapreduce.JobSubmitter: Submitting tokens for job: job_local91822813_0001
2025-05-06 14:59:23,488 INFO mapreduce.JobSubmitter: Executing with tokens: []
2025-05-06 14:59:23,560 INFO mapreduce.Job: The url to track the job: http://localhost:8080/
2025-05-06 14:59:23,560 INFO mapreduce.Job: Running job: job_local91822813_0001
2025-05-06 14:59:23,561 INFO mapred.LocalJobRunner: OutputCommitter set in config null
2025-05-06 14:59:23,564 INFO output.PathOutputCommitterFactory: No output committer factory defined, defaulting to FileOutputCommitterFactory
2025-05-06 14:59:23,565 INFO output.FileOutputCommitter: File Output Committer Algorithm version is 2
2025-05-06 14:59:23,565 INFO output.FileOutputCommitter: FileOutputCommitter skip cleanup _temporary folders under output directory:false, ignore cleanup failures: false
2025-05-06 14:59:23,565 INFO mapred.LocalJobRunner: OutputCommitter is org.apache.hadoop.mapreduce.lib.output.FileOutputCommitter
2025-05-06 14:59:23,602 INFO mapred.LocalJobRunner: Waiting for map tasks
2025-05-06 14:59:23,603 INFO mapred.LocalJobRunner: Starting task: attempt_local91822813_0001_m_000000_0
2025-05-06 14:59:23,615 INFO output.PathOutputCommitterFactory: No output committer factory defined, defaulting to FileOutputCommitterFactory
2025-05-06 14:59:23,615 INFO output.FileOutputCommitter: File Output Committer Algorithm version is 2
2025-05-06 14:59:23,615 INFO output.FileOutputCommitter: FileOutputCommitter skip cleanup _temporary folders under output directory:false, ignore cleanup failures: false
2025-05-06 14:59:23,622 INFO mapred.Task:  Using ResourceCalculatorProcessTree : [ ]
2025-05-06 14:59:23,624 INFO mapred.MapTask: Processing split: hdfs://localhost:9000/weather/test.txt:0+888190
2025-05-06 14:59:23,658 INFO mapred.MapTask: (EQUATOR) 0 kvi 26214396(104857584)
2025-05-06 14:59:23,658 INFO mapred.MapTask: mapreduce.task.io.sort.mb: 100
2025-05-06 14:59:23,658 INFO mapred.MapTask: soft limit at 83886080
2025-05-06 14:59:23,658 INFO mapred.MapTask: bufstart = 0; bufvoid = 104857600
2025-05-06 14:59:23,658 INFO mapred.MapTask: kvstart = 26214396; length = 6553600
2025-05-06 14:59:23,660 INFO mapred.MapTask: Map output collector class = org.apache.hadoop.mapred.MapTask$MapOutputBuffer
2025-05-06 14:59:23,743 INFO mapred.LocalJobRunner:
```