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

```
package temp;

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.FileInputFormat; import
org.apache.hadoop.mapreduce.lib.output.FileOutputFormat;

public class AverageDriver {

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

if (args.length != 2) {

System.err.println("Please Enter the input and output parameters");

System.exit(-1);

}

Job job = new Job(); job.setJarByClass(AverageDriver.class);
job.setJobName("Max temperature");

FileInputFormat.addInputPath(job, new Path(args[0]));
FileOutputFormat.setOutputPath(job, new Path(args[1]));
job.setMapperClass(AverageMapper.class);
job.setReducerClass(AverageReducer.class); job.setOutputKeyClass(Text.class);
job.setOutputValueClass(IntWritable.class);
System.exit(job.waitForCompletion(true) ? 0 : 1);   }

}
```

Mapper

```
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;

public void map(LongWritable key, Text value, Mapper<LongWritable, Text, Text, IntWritable>.Context context) throws IOException, InterruptedException {         int temperature;

String line = value.toString();      String year = line.substring(15, 19);      if (line.charAt(87) == '+') { temperature = Integer.parseInt(line.substring(88, 92));

} else {

temperature = Integer.parseInt(line.substring(87, 92));

}

String quality = line.substring(92, 93);

if (temperature != 9999 && quality.matches("[01459]")) context.write(new Text(year), new

IntWritable(temperature));

}

}
```

Reducer

```
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> {         public void reduce(Text key, Iterable<IntWritable> values, Reducer<Text, IntWritable, Text, IntWritable>.Context context) throws IOException, InterruptedException {    int max_temp = 0; int count = 0;

for (IntWritable value : values)
```

```
{         max_temp += value.get(); count++;     }
context.write(key, new IntWritable(max_temp / count));
}
}
```