**AIM:** To write a map reduce program to compute average of numbers in a file.

**PROGRAM:**

**/\*MAPPER\*/**

import java.util.\*;

import java.io.\*;

import org.apache.hadoop.io.\*;

import org.apache.hadoop.mapreduce.\*;

class AMapper extends Mapper<LongWritable,Text,Text,Text>

{

int i,sum,j;

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

{

String line=value.toString();

String[] num\_arr=line.split(" ");

sum=0;

for(i=0;i<num\_arr.length;i++)

{

j=Integer.parseInt(num\_arr[i]);

sum=sum+j;

}

contex.write(new Text("Avg"),new Text(sum+" "+i));

}

}

**/\*REDUCER\*/**

import java.util.\*;

import java.io.\*;

import org.apache.hadoop.io.\*;

import org.apache.hadoop.mapreduce.\*;

public class AReducer extends Reducer<Text, Text, Text, DoubleWritable>

{

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

{

double x,y,sum\_element=0,no\_of\_elem=0;

for(Text txt:values)

{

String element=txt.toString();

String[] each\_elem=element.split(" ");

for(int i=0;i<each\_elem.length;i++)

{

x=Double.parseDouble(each\_elem[0]);

y=Double.parseDouble(each\_elem[1]);

sum\_element=sum\_element+x;

no\_of\_elem=no\_of\_elem+y;

}

}

double avg=sum\_element/no\_of\_elem;

context.write(key,new DoubleWritable(avg));

}

}

**/\*DRIVER\*/**

import java.util.\*;

import java.io.\*;

import org.apache.hadoop.io.\*;

import org.apache.hadoop.mapreduce.\*;

import org.apache.hadoop.fs.Path;

import org.apache.hadoop.conf.Configuration;

import org.apache.hadoop.mapreduce.lib.input.FileInputFormat;

import org.apache.hadoop.mapreduce.lib.input.TextInputFormat;

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

import org.apache.hadoop.mapreduce.lib.output.TextOutputFormat;

public class ADriver

{

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

{

Configuration conf = new Configuration();

Job job = Job.getInstance(conf,"xyz");

job.setMapOutputKeyClass(Text.class);

job.setMapOutputValueClass(Text.class);

job.setOutputKeyClass(Text.class);

job.setOutputValueClass(DoubleWritable.class);

job.setMapperClass(AMapper.class);

job.setReducerClass(AReducer.class);

FileInputFormat.addInputPath(job, new Path(args[0]));

FileOutputFormat.setOutputPath(job, new Path(args[1]));

job.setJarByClass(ADriver.class);

job.waitForCompletion(true);

}

}

**OUTPUT:**

