

```

//Mario Diaz

import java.util.Scanner;

import java.io.File;
import java.io.FileWriter;
import java.io.IOException;

//https://www.w3schools.com/java/java_arraylist.asp
import java.util.ArrayList;

public class Main
{

    public static void main(String[] args)
    {
        ArrayList<TravelTimeData> data = new
ArrayList<TravelTimeData>();

        try
        {
            File f = new File("TravelDataPartial.csv");
            Scanner scan = new Scanner(f);
            //System.out.println(scan.nextLine());

            //Strip first line
            scan.nextLine();

            int index = 0;

            while(scan.hasNext())
            {
                //TODO: Lots of your code goes here
                //for file reading.

                String line = scan.nextLine();
                String[] values = line.split(",");

                TravelTimeData temp = new
TravelTimeData(Integer.parseInt(values[0]), values[1],
Integer.parseInt(values[values.length-1]));

                //System.out.println(scan.nextLine());
                //System.out.println(line);
                //System.out.println(values[15]);
                //System.out.println(values);
                line = scan.nextLine();
                values = line.split(",");
                temp.secondDirection( values [1],
Integer.parseInt( values[values.length-1]));

                data.add(temp);
            }
        }
    }
}

```

```

        } //While
        scan.close();
    }
    catch(Exception e)
    {
        System.out.println("ERROR!");
        e.printStackTrace();
    }

    TravelTimeData route = getLargestDifferential( data );
    System.out.println("\n");
    String temp = "The route with the biggest time differential
is\n"+route.toString()+"with a time difference of "+route.difference()+"
minutes.";

    System.out.println( temp );

    //write output file
    try
    {
        //You will need to change and fill in
        //the following.

        FileWriter fw = new FileWriter("output.txt");
        String stemp = String.format("%s %7s %4s %6s %4s\n",
"COGID", "Dir1", "TT1", "Dir2", "TT2");
        fw.write(stemp);

        for(int i=0; i<data.size(); i++)
        {
            TravelTimeData nd = data.get(i);
            fw.write(nd.toString());
        }
        fw.write("\n");
        fw.write(temp);
        fw.close();
    }
    catch(IOException e)
    {
        System.out.println(e);
    }
}

```

```

    public static TravelTimeData
getLargestDifferential( ArrayList<TravelTimeData> data)
    {
        int largest = 0;
        TravelTimeData toReturn = null;

        for( TravelTimeData ttd : data )

```

```

        {
            if( ttd.difference() > largest )
            {
                toReturn = ttd;
                largest = ttd.difference();
            }
        }
        return toReturn;
    }
}

```

//Mario Diaz

```

public class TravelTimeData
{
    private int COGID;
    private String direction1;
    private int travelTime1;
    private String direction2;
    private int travelTime2;

    //constructor
    public TravelTimeData(int COGID, String direction1, int
travelTime1)
    {
        this.COGID = COGID;
        this.direction1 = direction1;
        this.travelTime1 = travelTime1;
    }

    public int getCOGID(){ return COGID; }
    public String getDirection1(){ return direction1; }
    public int getTravelTime1(){ return travelTime1; }
    public String getDirection2(){ return direction2; }
    public int getTravelTime2(){ return travelTime2; }

    public void secondDirection (String direction2, int travelTime2)
    {
        this.direction2 = direction2;
        this.travelTime2 = travelTime2;
    }

    public int difference()
    {
        return Math.abs (travelTime1 - travelTime2);
    }
}

```

```

@Override
public String toString()
{
    return String.format("%s %6s %4s %6s %4d\n",
        COGID, direction1, travelTime1, direction2,
travelTime2);
    }
}

/* Bottom is CSV file */

/*
COGID,TravelDirection,RouteID,Route,Card,FromLocation,ToLocation,LengthIn
Miles,FunctionalClass,PostedSpeed,FreeFlowSpeed,AverageSpeed,TTI,FreeFlow
TravelTime,AverageTravelTime
10004,NB,45,N.M. 528,251,NORTH OF CORRALES RD. (NM 448),.118 MILES NORTH
OF CORRAL,0.123,Urban Principal Arterial,50,50.1,48.8,1.03,9,9
10004,SB,45,N.M. 528,251,NORTH OF CORRALES RD. (NM 448),.118 MILES NORTH
OF CORRAL,0.123,Urban Principal Arterial,50,52.5,49.7,1.06,8,9
10008,NB,45,N.M. 528,243,N.E. RIO VISTA,S.W. OF RIO ARRIBA,0.474,Urban
Principal Arterial,50,55,47.2,1.17,31,36
10008,SB,45,N.M. 528,243,N.E. RIO VISTA,S.W. OF RIO ARRIBA,0.474,Urban
Principal Arterial,50,55,47.1,1.17,31,36
10012,WB,321,NORTHERN BLVD.,11,EAST OF UNSER(NORTH),WEST OF
IDALIA,0.757,Urban Minor Arterial,40,45,36.1,1.25,61,76
10012,EB,321,NORTHERN BLVD.,11,EAST OF UNSER(NORTH),WEST OF
IDALIA,0.757,Urban Minor Arterial,40,40,37.2,1.08,68,73
10013,EB,321,NORTHERN BLVD.,40,EAST OF ROCKAWAY,N.W. OF
SARATOGA,1.038,Urban Minor Arterial,45,50,38.4,1.3,75,97
10013,WB,321,NORTHERN BLVD.,40,EAST OF ROCKAWAY,N.W. OF
SARATOGA,1.038,Urban Minor Arterial,45,50,35.2,1.42,75,106
10014,EB,321,NORTHERN BLVD.,24,EAST OF 40TH ST.,WEST OF
ROCKAWAY,0.348,Urban Minor Arterial,45,50,38.4,1.3,25,33
10014,WB,321,NORTHERN BLVD.,24,EAST OF 40TH ST.,WEST OF
ROCKAWAY,0.348,Urban Minor Arterial,45,50,35.2,1.42,25,36
10020,NB,45,N.M. 528,231,NORTH OF ROCKAWAY,SOUTH OF NORTHERN,0.681,Urban
Principal Arterial,45,45.1,40.6,1.12,54,61
10020,SB,45,N.M. 528,231,NORTH OF ROCKAWAY,SOUTH OF NORTHERN,0.681,Urban
Principal Arterial,45,50,35.8,1.41,49,69
10025,WB,321,NORTHERN BLVD.,15,EAST OF BROADMOOR,WEST OF LOMA
LARGA,1.029,Urban Minor Arterial,40,45,35.2,1.28,82,105
10025,EB,321,NORTHERN BLVD.,15,EAST OF BROADMOOR,WEST OF LOMA
LARGA,1.029,Urban Minor Arterial,40,41,38.3,1.07,90,97
10028,NB,45,N.M. 528,221,NORTH OF SUNDT,SOUTH OF ROCKAWAY,0.237,Urban
Principal Arterial,50,55,40.6,1.36,16,21
10028,SB,45,N.M. 528,221,NORTH OF SUNDT,SOUTH OF ROCKAWAY,0.237,Urban
Principal Arterial,50,55,35.8,1.55,16,24
10032,SB,45,N.M. 528,211,NORTH OF LEON GRANDE,SOUTH OF SUNDT,0.514,Urban
Principal Arterial,45,50,35.8,1.41,37,52
10032,NB,45,N.M. 528,211,NORTH OF LEON GRANDE,SOUTH OF SUNDT,0.514,Urban
Principal Arterial,45,45.1,40.6,1.12,41,46
10044,NB,45,N.M. 528,201,NORTH OF HIGH RESORT,SOUTH OF LEON
GRANDE,0.441,Urban Principal Arterial,45,45.1,40.6,1.12,35,39

```

10044,SB,45,N.M. 528,201,NORTH OF HIGH RESORT,SOUTH OF LEON
GRANDE,0.441,Urban Principal Arterial,45,50,35.8,1.41,32,45
10055,EB,321,NORTHERN BLVD.,12,EAST OF IDALIA,WEST OF
BROADMOOR,0.362,Urban Minor Arterial,40,40,37.2,1.08,33,35
10055,WB,321,NORTHERN BLVD.,12,EAST OF IDALIA,WEST OF
BROADMOOR,0.362,Urban Minor Arterial,40,45,36.2,1.25,29,36
10059,WB,416,SOUTHERN BLVD.,1,EAST OF IDALIA,WEST OF RAINBOW,0.838,Urban
Principal Arterial,35,43,40.8,1.06,70,74
10059,EB,416,SOUTHERN BLVD.,1,EAST OF IDALIA,WEST OF RAINBOW,0.838,Urban
Principal Arterial,35,44,43.4,1.01,69,69
10060,WB,416,SOUTHERN BLVD.,10,EAST OF ATLANTIC RD.,WEST OF
BALTIC,0.564,Urban Principal Arterial,35,43,40.8,1.06,47,50
10060,EB,416,SOUTHERN BLVD.,10,EAST OF ATLANTIC RD.,WEST OF
BALTIC,0.564,Urban Principal Arterial,35,44,43.4,1.01,46,47
10064,EB,416,SOUTHERN BLVD.,21,EAST OF BALTIC,WEST OF PECOS
LOOP,0.515,Urban Principal Arterial,40,45,35.7,1.26,41,52
10064,WB,416,SOUTHERN BLVD.,21,EAST OF BALTIC,WEST OF PECOS
LOOP,0.515,Urban Principal Arterial,40,45,36.6,1.23,41,51
*/

/*

INSTRUCTIONS:

Routes are distinguishable by their COGID. For a given COGID there are two TravelDirection's.

Identify the route with the largest differential between AverageTravelTime in the route's TravelDirection's.

Steps:

1. Read in the data.
2. Organize the data with a dedicated object and an ArrayList to hold instances of those objects.
3. Calculate maximum difference in AverageTravelTime.
4. Display results on command prompt and also output the results to a text file.

The correct answer is the route with COGID 10013. I displayed my results as follows and I recommend that you do the same:

The route with the biggest time differential is
10013 EB 97 WB 106
with a time difference of 9 minutes.

*/