

Name: Deepak Kumar  
CMSC 113 Assignment#1

#### Part 1

```
// File: Hours.java
// Written by: Deepak Kumar
// Date: September 18, 2018
// Purpose: Convert minutes into hours and minutes
// Examples/Usage:
//      $ java-introcs Hours 67
//      1 hours and 7 minutes
//      $ java-introcs Hours 453
//      7 hours and 33 minutes
//      $ java-introcs Hours 360
//      6 hours and 0 minutes
public class Hours {
    public static void main(String[] args) {
        // Input minutes
        int minutes = Integer.parseInt(args[0]);

        // Do calculations
        int hours = minutes / 60;
        minutes = minutes % 60;

        // Output
        System.out.println(hours + " hours and " + minutes + " minutes");
    } // main()
} // class main()
```

Sample Input/Output

```
A1> java-introcs Hours 67
1 hours and 7 minutes
```

```
A1> java-introcs Hours 453
7 hours and 33 minutes
```

```
A1> java-introcs Hours 360
6 hours and 0 minutes
```

## Part 2

```
// File: ArrivalTime.java
// Written by: Deepak Kumar
// Date: September 18, 2018
// Purpose: Computes Travel Time
// Examples/Usage:
//          $ java-introcs ArrivalTime 11 10 15 60
//          Arrival Time is 11:25
//          $ java-introcs ArrivalTime 15 50 125 50
//          Arrival Time is 18:20
//          $ java-introcs ArrivalTime 22 15 110 60
//          Arrival Time is 0:5
public class ArrivalTime {
    public static void main(String[] args) {
        // Inputs
        int h = Integer.parseInt(args[0]);
        int m = Integer.parseInt(args[1]);
        double d = Double.parseDouble(args[2]);
        double s = Double.parseDouble(args[3]);

        int t = (int) (d / s * 60);    // Travel time in minutes
        int minutes = h * 60 + m + t; // Arrival time in minutes
        h = minutes / 60 % 24;       // Arrival time (hours)
        m = minutes % 60;            // Arrival time (min)

        System.out.println("Arrival Time is " + h + ":" + m);
    } // main()
} // class ArrivalTime
```

Sample Input/Output

```
A1> java ArrivalTime 11 10 15 60
Arrival Time is 11:25
```

```
> java ArrivalTime 15 50 125 50
Arrival Time is 18:20
```

```
A1> java ArrivalTime 22 15 110 60
Arrival Time is 0:5
```

### Part 3

```
// File: GreatCircle.java
// Written by: Deepak Kumar
// Date: September 18, 2018
// Purpose: a Java program that takes four double
// command-line arguments—x1, y1, x2, y2—(the latitude and
// longitude, in degrees, of two points on the earth) and
// prints the great-circle distance between them.
// The great-circle distance, d (in nautical miles) is
// given by the following equation:
//
//  $d = E * \arccos(\sin(x1) * \sin(x2) + \cos(x1) * \cos(x2) * \cos(y1 - y2))$ 
//
// Where the earth's radius, E = 3986 miles.
// The distance, d will be miles.
//
// [xena@codewarrior ~]$ java GreatCircle 39.9526 75.1652 21.3069 157.8583
// 4945.290632589314 miles
```

```
public class GreatCircle {
    public static void main(String[] args) {
        // Input lat and long for source and destination
        double x1 = Math.toRadians(Double.parseDouble(args[0]));
        double y1 = Math.toRadians(Double.parseDouble(args[1]));
        double x2 = Math.toRadians(Double.parseDouble(args[2]));
        double y2 = Math.toRadians(Double.parseDouble(args[3]));
        double E = 3986.0; // Miles
        // double E = 6367.0; // KM

        double distance;
        distance = E * Math.acos(Math.sin(x1) * Math.sin(x2)
            + Math.cos(x1)
            * Math.cos(x2) * Math.cos(y1 - y2));

        // Output
        System.out.println(distance + " miles");
    } // main()
} // class GreatCircle
```

Sample Input/Output

```
A1> java-introcs GreatCircle 39.9526 75.1652 21.3069 157.8583
4945.290632589314 miles
```

```
A1> java-introcs GreatCircle 48.87 -2.33 37.8 122.4
5598.250822014901 miles
```