

```

/*
 * Temperatures2.java
 *
 * Created on Feb 11, 2004
 * Updated on May 2, 2005
 *
 * Author: Paolo Bucci
 *
 * Example of use of arrays. Reads a bunch of real numbers
 * (temperatures) and computes the average, the median, and
 * the number of temperatures above and below the average.
 */

import java.util.Scanner;

public class Temperatures2
{
    public static void main(String[] args)
    {
        Scanner keyboard = new Scanner(System.in);

        int N = keyboard.nextInt(); // input number of temperatures
        double [] temperatures = new double[N];
        double sum = 0.0;

        // input the temperatures into the array and compute the sum
        for (int i = 0; i < N; i++)
        {
            temperatures[i] = keyboard.nextDouble();
            sum = sum + temperatures[i];
        }

        // compute the average
        double average = sum / N;

        // count the number of temperatures above and below the average
        int below = 0, above = 0;
        for (int i = 0; i < N; i++)
        {
            if (temperatures[i] < average)
            {
                below++;
            }
            else if (temperatures[i] > average)
            {
                above++;
            }
        }

        // order the temperatures in increasing order
        sort(temperatures);

        // compute the median of the temperatures
        double median;
        if (N % 2 == 0) // N is even
        {
            median = (temperatures[N/2-1] + temperatures[N/2]) / 2.0;
        }
        else // N is odd
        {

```

```

        median = temperatures[N/2];
    }

    // output the results
    System.out.println("average = " + average);
    System.out.println("above = " + above);
    System.out.println("below = " + below);
    System.out.println("median = " + median);
}

/*
 * Sort the temperatures array in increasing order.
 */
private static void sort(double[] temps)
{
    for (int index = 0; index < temps.length; index++)
    {
        int indexOfNextSmallest = indexOfSmallest(index, temps);
        interchange(index, indexOfNextSmallest, temps);
    }
}

/*
 * Find and return the index of the smallest value in array t
 * starting at index startIndex.
 */
private static int indexOfSmallest(int startIndex, double[] t)
{
    double min = t[startIndex];
    int indexOfMin = startIndex;
    for (int index = startIndex + 1; index < t.length; index++)
    {
        if (t[index] < min)
        {
            min = t[index];
            indexOfMin = index;
        }
    }
    return indexOfMin;
}

/*
 * Swaps the elements at positions i and j in array t.
 */
private static void interchange(int i, int j, double[] t)
{
    double tmp = t[i];
    t[i] = t[j];
    t[j] = tmp;
}
}

```