

```
// ***** Lab1.c *****

// Program written by: Daniel Canterino

// Date Created: 1/18/2017

// Last Modified: 2/1/17

// Brief description of the Lab

// An embedded system is capturing temperature data from a
// sensor and performing analysis on the captured data.

// The controller part of the system is periodically capturing N
// readings of the temperature sensor. Your task is to write three
// analysis routines to help the controller perform its function

// The three analysis subroutines are:

// 1. Calculate the mean of the temperature readings
//    rounded down to the nearest integer

// 2. Calculate the range of the temperature readings,
//    defined as the difference between the largest
//    and smallest reading

// 3. Check if the captured readings are a non-increasing montonic series
//    This simply means that the readings are sorted in non-increasing order.
//    We do not say "increasing" because it is possible for consecutive values
//    to be the same, hence the term "non-increasing". The controller performs
//    some remedial operation and the desired effect of the operation is to
//    lower the the temperature of the sensed system. This routine helps
//    verify whether this has indeed happened

#include <stdint.h>

#define True 1
```

```

#define False 0

#define N 21    // Number of temperature readings

uint8_t Readings[N]; // Array of temperature readings to perform analysis on


// Return the computed Mean

uint8_t Find_Mean(){

// Replace ths following line with your solution

    int i;          //initializes variables for mean

    int sum=0;

    int mean=0;

    for (i=0; i<N; i++){

        sum=sum+Readings[i]; //calculates the sum of the integers in the array

    }

    mean=sum/N; //calculates mean by dividing sum by the number of elements in the sum

    return(mean); //returns the mean

}


// Return the computed Range

uint8_t Find_Range(){

// Replace ths following line with your solution

    int range=0; //initializes variables

    int i;

    int max=0;

    int min=Readings[0]; //sets min to non-0 element, next lines iterate through the entire array,
    finding the max and the min value

```

```

    for (i=0; i<N; i++){
        if (Readings[i]>max){
            max=Readings[i];
        }
        if (Readings[i]<min){
            min=Readings[i];
        }
    }

    range=max-min;//finds range by subtracting the min value from the max
    return(range);//returns the range
}

```

// Return True or False based on whether the readings

// a non-increasing montonic series

```
uint8_t IsMonotonic(){
```

// Replace the following line with your solution

```

    int i;

    int reading;//initializes variables

    for (i=0; i<N; i++){//iterates through the entire array, compares if the next value is less than or
    equal to the current one

        if (Readings[i]>=Readings[i+1]){

            reading=True;//sets reading to true if this is the case

        }

        if (Readings[i]<Readings[i+1]){

            reading=False;//if not, it sets reading to false and breaks

            break;

```

```
        }  
    }  
  
    return(reading); //returns the reading  
}
```

//Testcase 0:

// Scores[N] = {80,75,73,72,90,95,65,54,89,45,60,75,72,78,90,94,85,100,54,98,75};

// Range=55 Mean=77 IsMonotonic=False

//Testcase 1:

// Scores[N] = {100,98,95,94,90,90,89,85,80,78,75,75,75,73,72,72,65,60,54,54,45};

// Range=55 Mean=77 IsMonotonic=True

//Testcase 2:

// Scores[N] = {80,80};

// Mean=80 Range=0 IsMonotonic=True

//Testcase 3:

// Scores[N] = {100,80,40,100,80,40,100,80,40,100,80,40,100,80,40,100,80,40,100,80,40};

// Mean=73 Range=60 IsMonotonic=False

//Testcase 4:

// Scores[N] = {100,95,90,85,80,75,70,65,60,55,50,45,40,35,30,25,20,15,10,5,0};

// Range=100 Mean=50 IsMonotonic=True

UART #1

```
Temperature Sensor Data Analysis
Test Case 0
ate Yes, Your Mean= 77
Yes, Your Range= 55
Correct Analysis of monotonicity
Test Case 1
Yes, Your Mean= 77
Yes, Your Range= 55
Correct Analysis of monotonicity
Test Case 2
= ma: Yes, Your Mean= 80
Yes, Your Range= 0
Correct Analysis of monotonicity
Test Case 3
Yes, Your Mean= 73
Yes, Your Range= 60
Correct Analysis of monotonicity
Test Case 4
Yes, Your Mean= 50
Yes, Your Range= 100
Correct Analysis of monotonicity
= if Passed all tests - End of Analysis
```

Call Stack + Locals

UART #1

Memory 1