

```

1 // ***** Lab1.c *****
2 // Program written by: put your names here
3 // Date Created: 1/18/2017
4 // Last Modified: 1/18/2017
5 // Brief description of the Lab
6 // An embedded system is capturing temperature data from a
7 // sensor and performing analysis on the captured data.
8 // The controller part of the system is periodically capturing N
9 // readings of the temperature sensor. Your task is to write three
10 // analysis routines to help the controller perform its function
11 // The three analysis subroutines are:
12 // 1. Calculate the mean of the temperature readings
13 // rounded down to the nearest integer
14 // 2. Calculate the range of the temperature readings,
15 // defined as the difference between the largest
16 // and smallest reading
17 // 3. Check if the captured readings are a non-increasing montonic series
18 // This simply means that the readings are sorted in non-increasing order.
19 // We do not say "increasing" because it is possible for consecutive values
20 // to be the same, hence the term "non-increasing". The controller performs
21 // some remedial operation and the desired effect of the operation is to
22 // lower the the temperature of the sensed system. This routine helps
23 // verify whether this has indeed happened
24 #include <stdint.h>
25 #define True 1
26 #define False 0
27 #define N 21 // Number of temperature readings
28 uint8_t Readings[N]; // Array of temperature readings to perform analysis on
29
30 // Return the computed Mean
31 uint8_t Find_Mean(){
32     uint8_t i = 0;
33     uint32_t a = 0;
34     for(i = 0; i < 21; i++)
35         a += Readings[i];
36     a /= 21;
37     return(a);
38 }
39
40 // Return the computed Range
41 uint8_t Find_Range(){
42 // Replace ths following line with your solution
43     uint8_t max = 0;
44     uint8_t min = 0;
45     uint8_t c = 0;
46     uint8_t foo = 0;
47     uint8_t range = 0;
48     max = Readings[0];
49     min = Readings[0];
50     for(c = 1; c < 21; c++)
51     {
52         foo = Readings[c];
53         if(foo > max)
54             max = Readings[c];
55         else if(foo < min)
56             min = Readings[c];
57     }
58     range = max - min;
59     return(range);
60 }
61
62 // Return True of False based on whether the readings
63 // a non-increasing montonic series
64 uint8_t IsMonotonic(){
65     uint8_t aValue = 0;
66     uint8_t bValue = 0;
67     for(int f = 0; f < 20; f++)
68     {
69         aValue = Readings[f];
70         bValue = Readings[f+1];
71         if(aValue < bValue)
72             return(False);

```

```
73     }
74     return (True);
75 }
76
77 //Testcase 0:
78 // Scores[N] = {80,75,73,72,90,95,65,54,89,45,60,75,72,78,90,94,85,100,54,98,75};
79 // Range=55 Mean=77 IsMonotonic=False
80 //Testcase 1:
81 // Scores[N] = {100,98,95,94,90,90,89,85,80,78,75,75,75,73,72,72,65,60,54,54,45};
82 // Range=55 Mean=77 IsMonotonic=True
83 //Testcase 2:
84 // Scores[N] = {80,80,80,80,80,80,80,80,80,80,80,80,80,80,80,80,80,80,80,80,80};
85 // Mean=80 Range=0 IsMonotonic=True
86 //Testcase 3:
87 // Scores[N] = {100,80,40,100,80,40,100,80,40,100,80,40,100,80,40,100,80,40,100,80,40};
88 // Mean=73 Range=60 IsMonotonic=False
89 //Testcase 4:
90 // Scores[N] = {100,95,90,85,80,75,70,65,60,55,50,45,40,35,30,25,20,15,10,5,0};
91 // Range=100 Mean=50 IsMonotonic=True
92
93
```