```c
// ***************** Lab2.c ***************
// Program written by: Roberto Reyes
// Date Created: 1/18/2017
// Last Modified: 2/10/2020
// 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 size
// 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. Convert from Centigrate to Farenheit using integer math
//    3. Check if the captured readings are a non-increasing monotonic series
//       This simply means that the readings are sorted in non-increasing order.
//       We do not say "decreasing" 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
//       raise 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

// Return the computed Mean
// Inputs: Readings is an array of 16-bit temperature measurements
//         N is the number of elements in the array
// Output: Average of the data
// Notes: you do not need to implement rounding
int16_t Find_Mean(int16_t const Readings[], int32_t const N){
// Replace this following line with your solution
  int32_t total = 0;
  int16_t i = 0;
  while(i < N){
    total += Readings[i];
    i++;
  }
  int16_t avg = total/N;
  return avg;
}

// Convert temperature in Centigrade to temperature in Farenheit
// Inputs: temperature in Centigrade
// Output: temperature in Farenheit
// Notes: you do not need to implement rounding
int16_t CtoF(int16_t const TinC){
// Replace this following line with your solution
  int16_t F = (TinC*9)/5+32;

  return F;
}

// Return True of False based on whether the readings
// are an increasing monotonic series
// Inputs: Readings is an array of 16-bit temperature measurements
// N is the number of elements in the array
// Output: true if monotonic decreasing, false if nonmonotonic
int IsMonotonic(int16_t const Readings[], int32_t const N){
// Replace this following line with your solution
 if(N==1) return True;
 else{
   int16_t i = 1;
   while (i < N){
     if(Readings[i-1] < Readings[i]) return False;
     else i++;
   }

 return True;

}
```

```
73    }
74
75
```