

Assignment 02

CSE 241

April 6, 2022

1 Description

You are to write a C++ program which reads in a list of temperatures from standard input (`cin`). You must read in all the temperatures and store them in an array. You may assume there will never be more than 1000 temperatures. This is important, as you will be forced to declare an array of fixed size. You may not use a list or vector for this assignment.

If there is bad input for one number, store the special value Not-A-Number in the array at the given position, then continue reading further values. Not-A-Number can be stored by using the `NAN` macro defined in `cmath`, e.g.:

```
x[i] = NAN;
```

Later, you may test if a number is NAN using the special `isnan` function:

```
if (isnan(x[i])) {  
    // ...  
}
```

Note that checking with, e.g., `x[i] == NAN` **will not work**.

Use `cin.eof()` (end-of-file) to determine when you have read in every value. Although you can assume there are no more than 1000 values, you cannot make an assumption about the exact number.

Your output will come in 3 parts:

1. A list of “Daily highs”. Since each temperature reading represents 1 hour, you must iterate through your array 24 values at a time. For each 24 value block, calculate the maximum value and print it out.
2. A “Sliding window of 24-hour averages”. Iterate through each temperature reading, one at a time. Calculate the mean of all temperatures from 12 hours prior to 11 hours ahead. When near the beginning or end of the array, you must be careful not to overrun the bounds of the array. E.g., if for hour #3, calculate the average of hours #0-#14. Print out a two-column table, with the raw temperature reading in the left column and the sliding window 24-hour average in the right column.
3. The mean temperature of *all* readings, followed by the mean temperature of all readings *except* the highest, followed by the mean temperature of all readings except the *two* highest. Hint: use the `maximum_value` function to find the highest value and then set it to `NAN`.

All temperatures must be printed out with 3 digits after the decimal point. All tables of numbers must be right-aligned.

2 Files

2.1 array.h

Here are the contents of `array.h`. Do not change `array.h`.

```
1 #ifndef ARRAY_H
2 #define ARRAY_H
3
4 #include <cstdlib> // for a definition of size_t
5
6 double mean(double const*, size_t);
7 double* maximum_value(double*, size_t);
8
9 #endif
```

The `mean` function will calculate the mean of all numbers *ignoring Not-A-Numbers*. For example, if you call the function with the numbers [4, 6, NAN, 2, NAN], return a value of 4.0.

The `maximum_value` function must return a *pointer* to the largest value in the array. Do *not* return the largest value in the array, but a *pointer* to it.

2.2 array.cpp

Implement the `mean` and `maximum_value` functions.

2.3 main.cpp

Read in all values from `cin`. Perform all analysis and produce output.

3 Requirements

3.1 Building

For this assignment, it is required that your project build correctly using a Makefile. You will receive a grade of 0% if your project does not build correctly when running `make`.

3.2 Testing

The files, `sample-input.txt` and `sample-correct-output.txt` are provided for your convenience. More sample input or output data may be provided. TAs may use test cases not provided to students when testing assignments.