

# Assignment 1

50 points

## Purpose

The purpose of this assignment is to get you comfortable with the the Unix development environment and in creating makefiles.

## Assignment

This assignment prints out a table of wind chill temperatures for selected ranges of air temperatures and wind speeds.

In 2001, the National Weather Service adopted a new formula for calculating the wind chill felt by human beings in cold weather with a wind. The old formula, calculated in 1945, was based on how fast a can of water froze when hoisted 33 feet up a pole. But a human being is not a can of water and none of us are 33 feet tall, so the accuracy left something to be desired. The new formula is based on measuring how much heat is lost through the human face in various conditions and is much more accurate.

The new formula is:

$$\text{Wind Chill } (^{\circ}\text{F}) = 35.74 + 0.6215T - 35.75(V^{0.16}) + 0.4275T(V^{0.16})$$

where T is the air temperature in degrees Fahrenheit and V is the wind speed in miles per hour. This formula is considered accurate only for temperatures below 50°F and wind speeds above 3 mph.

## Program

Your program should ask the user for selected ranges of temperatures and wind speed values for which to calculate the wind chill. You should ask for starting and ending values for each parameter. You should also ask for a value by which each parameter will be incremented or decremented.

Temperature ranges start with a high value and go down. Wind speed ranges start with a low value and go up. You should perform error checking to enforce these constraints. If you have illegal values, print an error message and exit the program (use `return`). Increment and decrement values should also be positive and checked for errors.

All floating point values should be printed out with three decimals of precision after the decimal point. Trailing zeros should be printed. Each floating point value in the table should be right justified in a field that is nine characters wide.

Your program should have two functions (in addition to `main()`). One should calculate and return the wind chill value from the air temperature and wind speed passed in. Another function should print the

table given the start, end, and increment values for the temperature and wind speed. This function will call the function that calculates the wind chill.

The `main()` function should simply input the values, check for errors and call the table printing function.

The constraint that the wind chill formula is valid only for wind speeds greater than 3 mph and air temperatures less than 50 degrees should be built into the function that calculates wind chill. If either parameter is out of range, simply return the air temperature as the wind chill value. No error message or other handling is needed for this function.

## Output

Sample output from this program on `turing/hopper` is found below. The TAs will use other values to check your program.

```
hopper:$ assign1
This program prints out a table of selected wind chill factors

Temperature range high value (degrees F) ? 40
Temperature range low value  (degrees F) ? 0
Temperature decrement (degrees F) ? 6
Wind speed low value (mph) ? 5
Wind speed high value(mph) ? 60
Wind speed increment (mph) ? 3

Wind Chill Table

Air Temperature
(deg. F)    40.000    34.000    28.000    22.000    16.000    10.000    4.000
-----
Wind speed
(mph)
5.000| 36.472    29.425    22.378    15.330     8.283     1.236    -5.812
8.000| 34.588    27.282    19.975    12.668     5.362    -1.945    -9.251
11.000| 33.228    25.735    18.241    10.748     3.254    -4.239   -11.733
14.000| 32.152    24.510    16.868     9.227     1.585    -6.057   -13.698
17.000| 31.254    23.489    15.724     7.959     0.194    -7.571   -15.337
20.000| 30.481    22.609    14.738     6.867    -1.005    -8.876   -16.748
23.000| 29.800    21.835    13.870     5.904    -2.061   -10.026   -17.991
26.000| 29.190    21.141    13.092     5.043    -3.006   -11.055   -19.104
29.000| 28.636    20.511    12.386     4.260    -3.865   -11.990   -20.115
32.000| 28.128    19.934    11.739     3.544    -4.651   -12.846   -21.041
35.000| 27.660    19.400    11.141     2.881    -5.378   -13.638   -21.897
38.000| 27.223    18.904    10.584     2.265    -6.054   -14.374   -22.693
41.000| 26.815    18.439    10.064     1.688    -6.687   -15.063   -23.438
44.000| 26.431    18.003     9.574     1.146    -7.282   -15.711   -24.139
47.000| 26.069    17.590     9.112     0.634    -7.844   -16.323   -24.801
50.000| 25.725    17.200     8.674     0.149    -8.377   -16.902   -25.428
53.000| 25.398    16.828     8.258    -0.313    -8.883   -17.454   -26.024
56.000| 25.087    16.474     7.860    -0.753    -9.366   -17.979   -26.593
59.000| 24.789    16.135     7.481    -1.173    -9.828   -18.482   -27.136
```

```
hopper:$ assign1
This program prints out a table of selected wind chill factors
```

Temperature range high value (degrees F) ? 0  
 Temperature range low value (degrees F) ? -30  
 Temperature decrement (degrees F) ? 5  
 Wind speed low value (mph) ? 5  
 Wind speed high value (mph) ? 60  
 Wind speed increment (mph) ? 2

#### Wind Chill Table

Air Temperature (deg. F)	0.000	-5.000	-10.000	-15.000	-20.000	-25.000	-30.000
-----							
Wind speed (mph)							
5.000	-10.510	-16.383	-22.256	-28.128	-34.001	-39.874	-45.747
7.000	-13.068	-19.094	-25.120	-31.145	-37.171	-43.197	-49.223
9.000	-15.071	-21.216	-27.362	-33.507	-39.653	-45.798	-51.944
11.000	-16.729	-22.973	-29.218	-35.462	-41.707	-47.952	-54.196
13.000	-18.150	-24.479	-30.809	-37.139	-43.468	-49.798	-56.127
15.000	-19.398	-25.802	-32.206	-38.611	-45.015	-51.419	-57.823
17.000	-20.513	-26.984	-33.455	-39.926	-46.397	-52.868	-59.339
19.000	-21.523	-28.055	-34.586	-41.117	-47.648	-54.180	-60.711
21.000	-22.448	-29.034	-35.621	-42.207	-48.794	-55.380	-61.967
23.000	-23.301	-29.938	-36.576	-43.214	-49.851	-56.489	-63.126
25.000	-24.094	-30.779	-37.464	-44.149	-50.834	-57.519	-64.204
27.000	-24.835	-31.564	-38.294	-45.023	-51.752	-58.482	-65.211
29.000	-25.532	-32.303	-39.074	-45.845	-52.615	-59.386	-66.157
31.000	-26.189	-32.999	-39.809	-46.620	-53.430	-60.240	-67.050
33.000	-26.812	-33.659	-40.507	-47.354	-54.201	-61.049	-67.896
35.000	-27.403	-34.286	-41.169	-48.052	-54.935	-61.817	-68.700
37.000	-27.967	-34.884	-41.800	-48.717	-55.633	-62.550	-69.467
39.000	-28.506	-35.455	-42.404	-49.352	-56.301	-63.250	-70.199
41.000	-29.022	-36.002	-42.981	-49.961	-56.941	-63.920	-70.900
43.000	-29.518	-36.527	-43.536	-50.545	-57.555	-64.564	-71.573
45.000	-29.994	-37.032	-44.069	-51.107	-58.145	-65.183	-72.220
47.000	-30.453	-37.518	-44.583	-51.649	-58.714	-65.779	-72.844
49.000	-30.896	-37.987	-45.079	-52.171	-59.262	-66.354	-73.446
51.000	-31.324	-38.441	-45.558	-52.675	-59.793	-66.910	-74.027
53.000	-31.738	-38.880	-46.022	-53.164	-60.306	-67.448	-74.590
55.000	-32.139	-39.305	-46.471	-53.637	-60.803	-67.969	-75.135
57.000	-32.528	-39.717	-46.906	-54.096	-61.285	-68.474	-75.663
59.000	-32.906	-40.117	-47.329	-54.541	-61.753	-68.965	-76.177

## Other Points

- This assignment is worth 50 points. Future assignments will be worth 100 points. This assignment is much easier than all other assignments will be.
- The `pow()` or `powf()` function will be needed for this assignment. It can be found in `<cmath>`. You will need to look at the man page to determine how it is used. You will also need to link the math library into your program. This is done by placing the compiler directive `-lm` at the end of the command line command that you use to compile your program, i.e.,  
`g++ ... your arguments here ... -lm.`
- Create a directory specifically for this assignment. Call it "assign1".
- The name of your source code file should be `assign1.cc`.
- A makefile is required for this assignment. Please call your makefile "Makefile"

- You should have a makefile rule to clear out the project to a pristine state.
- Your `Makefile` should use appropriate variables.
- Don't forget your documentation, which will be minimal for this assignment.
- Symbolic constants should be used to avoid magic numbers. You may use either `#define` or `const`, whichever you know best at this point. You do not have to create symbolic constants for the coefficients used in the wind chill formula. Those are truly magic numbers that should never, ever be changed.
- Function prototypes are required for all functions you write (except `main()` of course)
- Programs that do not compile on `turing/hopper` automatically receive 0 points.

## Submission

Place your source code file and `Makefile` in a directory name `assign1`. Use the `mailprog` script on this directory as described on the course web site to submit the assignment.