Cat, Cot, Cap Temperature Calculator

Professor Caleb Fowler

1/10/24

# Problem.

Your friend has asked you to help them with a project. He needs to convert a Fahrenheit temperature to either Celsius, Kelvin or Rankine temperature scales. However, your friend is a bit - odd. He needs this to apply to only three objects: a cat, a cot and a cap. Each object has a valid temperature range for his calculations (See Table 1 below). Prompt the client for the name of the object to convert, then display a menu where they can choose the temp scale to convert to, then prompt for the object’s current temperature, finally, display the converted value.

# Requirements.

* Prompt for the object to calculate the temperature for. Note: This is NOT a menu! The client is expected to enter the proper words.
* Ask for the temperature of the object in Fahrenheit. Check to see if it’s in the proper range. Reprompt if incorrect.
* Create a menu of valid temperature scales to select from. Selecting the scale option will immediately display the converted temperature. Include an option to calculate the temperature for ALL the options. See **Table 1** for the allowable temperature ranges.
* Re-prompt if incorrect data was entered. Keep re-prompting until the client get’s it right.
* Include a void ProgramGreeting() function. This will run automatically once when the program ﬁrst starts. This function should display (on individual lines):

– A welcome message.

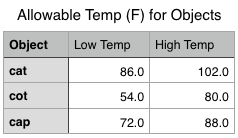
– Your name (as author).

– System date. Format this as MonthName day, year. Example:

June 30, 1988.

* Store the results of all computations.
* Include your speciﬁcation comments above the main portion of your code where you implement the speciﬁcation. No credit if this is missing!
* Use white-space and comments to make your code more readable.
* Use function prototypes for all functions.
* Do not use c (.h) style libraries. Use C++ libraries instead.
* Your program must compile in C++ on Ubuntu.
* Your program must generate logically correct output.
* Program activities are split into logical ’chunks’ or paragraphs. I’m expecting paragraphs for input, processing (if any), and output operations.
* If there is non integer output, force the computer to always display 3 places to the right of the decimal.
* Include a **Source File Header**, like you used in homework 1. Do not include a specification with it like you did in that assignment.
* Include a **Commented Sample Run** for this assignment like you did for homework 1. Again, do not include a specification with it.
* Include a **ProgramGreeting** for this assignment like you did for homework 1, but turn it into a function you call at the very start of main().

**Table 1. Example Fahrenheit Scale**



You will need to create similar tables   
for Celsius, Kelvin and Rankine scales.

# Speciﬁcations.

**// Speciﬁcation C1 – Only Valid Words**

We only want to accept valid words cat, cap, cot no matter how they are spelled (ie Cat, CAT, caT all Ok too). Put this in a function along with the text prompt. This function will only return an integer **objCode** corresponding to one of the following values: 0 - bad input, 1 - cat, 2 - cot, 3 - cap, 4 - free for your word (if you program one). You can use objCode in this function. You must use **objCode** in the function you call this from as the return variable. Use a do .. while loop to reprompt until correct.

**// Speciﬁcation C2 – Select Temp Scale Menu**

Create a menu which shows us what temperature scales we can convert to (Fahrenheit, Celsius, etc.). This can be a numeric menu (each option is selected by pressing a number). The client selects what scale they want to use for output. Remember to include an option to convert to all the scales.

**// Speciﬁcation C3 – Three Functions**

Include at least three functions in your program. Put this specification comment above your function prototype(s).

**// Speciﬁcation B1 – Display Function Activity**

Write to the terminal an announcement when every function is called. Normally, this would go into a log file, but we haven’t learned that yet. Color the text of these messages orange. An example might be something like:

Program Start

Function main() started

Function ProgramGreeting() started

This is my program greeting…

**// Speciﬁcation B2 – Valid Temps Only**

Make sure the user can input the temperatures only within a valid range. If out of range, tell them (too high, too low) and re-prompt. We can accept an error of less than +/-0.2 degrees. Example: a valid temperature for a cat will be 102.2, but 102.3 will not. Create a function **bool ValFlo()** which accepts the user entered ﬂoat value and returns a boolean value; true if the ﬂoat it’s checking is good and false if it isn’t.

**// Speciﬁcation B3 - Overloaded Prompt Functions**

Create a function called Prompt like the last assignment. Continue to pass the string to display when Prompt is called. This time, overload this function so 1 version of Prompt returns an integer, another a ﬂoat, and the third a string. Use any input mechanism you wish. The integer version can be used for the temperature scale selection menu, while the float version can be used for the temp value itself, for example. You can use the string anywhere appropriate.

**// Specification A1 - All Temps Table**

Generate a figure similar to table 1, but show the high and low values for all 4 temperature scales for all 3 objects. This could be useful for debugging because it will show all the temp values in one place. If you added a 4th object from C3, make sure you include it here as well.

**// Specification A2 - All Temp Conversion**

Add an option for any cat, cap, or cot temp value to convert to all available temperature scales. Display the output of this in a table with an appropriate header.

**// Specification A3 - Stack**

You are free to use a stack data structure for this assignment. It can be a global variable for this assignment as well! Remember, it works by Last In- First Out. You can use as many stacks as you want with whatever type(s) you want (a string stack may be useful for A2, for example).