

Computer Science 2211b
Software Tools and Systems Programming
Winter 2024

Assignment 2 (7%)

Due: Monday February 5 at 11:55 PM

Objectives

In this assignment, you will gain experience with:

- Using basic C types, statements, and arithmetic expressions
- Writing simple functions
- Following style conventions and good programming practices

PART 1 - Converter (60%)

Write a program `converter.c` that meets the requirements below:

1. Present a menu to the user with 4 options:
 - **1** for conversion between Celsius and Fahrenheit
 - **2** for conversion between Kilometers and Miles
 - **3** for conversion between Joules to Calories (1 cal = 4.184 J)
 - **0** to quit
 - For any other integer, print an error and display the menu again
2. Each action (1 to 3) should be implemented as a function. After the function is called, return to Step 1 to print the menu again. If 0 is selected, the program exits.
3. For actions 1 to 3, prompt the user for the direction of the conversion by entering a single character as shown below:
 - **Option 1:**
 - **C**: Celsius to Fahrenheit
 - **F**: Fahrenheit to Celsius
 - Any other input: print an error and repeat the prompt

- **Option 2:**
 - **K:** Kilometers to Miles
 - **M:** Miles to Kilometers
 - Any other input: print an error and repeat the prompt
 - **Option 3:**
 - **J:** Joules to Calories
 - **C:** Calories to Joules
 - Any other input: print an error and repeat the prompt
4. The program should then prompt for the value to convert. It then performs the conversion and displays the result
- You do not need to handle erroneous input. We will assume the user always enters valid numbers
 - The input value should be a floating-point number
 - The output should be displayed with **units** and **two decimal places** in a descriptive manner (e.g., $0.00C = 32.00F$).
5. Your program must follow the **CS 2211 Style Guide** introduced in Topic 5, including:
- Snake case for identifiers (variables and function names)
 - 4 spaces per indentation level -- no tabs
 - Clear comments throughout the program
 - Commented variable declarations
 - A file header comment
 - Function header comments indicating the purpose of each function
 - The use of constants rather than magic numbers

Hint: Your editor can be set to expand tabs to spaces and can be configured to use a specific indentation level. You can Google how to do this for your preferred editor.

Hint: When reading a character from the user with `scanf`, your program should handle leading whitespace characters including spaces, tabs, newlines, etc. Rather than using:

```
scanf("%c", &choice);
```

You will instead likely find the following useful:

```
scanf(" %c", &choice);
```

PART 2 - Statistics (40%)

Write a program `stats.c` that meets the requirements below:

1. Prompt the user to enter up to 10 digits (from 0 to 9)
2. At any time, the user should be able to type q to stop entering digits
3. Once 10 digits have been entered (or the user presses q), display the sum, count, and average (2 decimal places) of the digits entered. Example:

```
Enter a digit (q to quit): 1
Enter a digit (q to quit): 2
Enter a digit (q to quit): 3
Enter a digit (q to quit): q
Sum      : 6
Count    : 3
Average  : 2.00
```

4. Your program should adhere to the same CS 2211 Style Guide rules discussed in Part 1.

Submitting Your Assignment

After completing this assignment, your `asn2` directory should look as follows:

```
$ tree ~/courses/cs2211/asn2
.
├── converter.c
├── stats.c
└── accommodation.txt      (if you requested academic accommodation)

1 directory, 3 files
```

If not, go back and ensure that your directory matches this structure.

For full details on submission, see the **Assignment Submission Instructions** from the course web site. As a quick overview, submitting will involve the following commands:

```
cd ~/courses/cs2211/asn2
git add .
git commit -m "Submitting assignment 2"
git push
git tag asn2-submission
git push --tags
```

Check your repository and confirm that you have pushed your files successfully.

Then, check your Western email for an email confirmation from the system. If you don't receive one, check your junk mail folder. If you still don't receive an email within 10-15 minutes, follow the steps to resubmit your assignment.

This involves first deleting your `asn2-submission` tag:

```
git tag -d asn2-submission
git push origin :refs/tags/asn2-submission
```

Then, resubmit:

```
git tag asn2-submission
git push --tags
```