Recap SSH and Compilation

- `ssh hktang@ug24.cs.ualberta.ca` -- obviously, try this out with your own username
- Check out `declaration.c` to show an example of an error (need to compile the program to see the error)
- Fix error by adding a declaration of x
- Since x has not been initialized, compiling the program would now give a warning
- Fix the warning by adding a definition for x and try compiling again

Demonstrating Conversion Specifiers

- `emacs tprintf.c` to show program (can also use cat if you are not interested in editing)
- Compile program (remember that we compile using `gcc -Wall -std=c99 ...`). Look at Lecture 1 slides if you are not still sure how to compile
- Open another terminal to run the program in so you can see the source code and output side by side
- Compare each number in the output and the corresponding conversion specifier to understand the details

Demonstrating Escape Characters

- Look at the source code of `special_print.c` (you should know how to do that by now on the command line)
- See how each special character affected the output
- You will need to turn sound on/up to hear `\a`

Demonstrating Return Value of scanf

- Compile and run `celsius.c`
- Enter a fahrenheit temp when prompted
- Try any non-number
- Why did the program still "work" when you entered a non-number?
- Compile and run `celsius_scanf.c`
- Enter a fahrenheit temp when prompted
- Try any non-number
- Why did the program quit when you entered a non-number?

Additional Examples

These were not necessarily all demoed in class. Try them out on your own! Watch out for warnings and errors and see how you can fix those. After fixing any warnings or errors, compile and run these programs and think about the output and the behavior of these programs.

- `macro.c`
- `printf_format.c`
- `addfrac.c`
- `scanf_whitespace.c` -- try running this with different inputs and different spaces between these inputs. Observe how the program behaves.
- `scanf_matching.c`

Full Program for Computing Dimensional Weight

See the complete version of the program from the lecture slides: `weight-scanf.c` . Please note that in order to use the `ceil` function, you need to compile your code using `-lm` : `gcc -Wall -std=c99 -o weight-scanf weight-scanf.c -lm` . Note that the `-lm` option needs to appear after the source file name.