Intro. Computing with the C Programming Language

Functions

- PIC: Ch. 7. Working with Functions
- GCM: Ch. 5. Function

Shin Hong

16 November 2023

Defining a function

- The definition or declaration of a function must appear before the function is called

- If not explicitly given, the return value of a function is `int`

- A function receives zero, one, or multiple, or a variable number of arguments
  - Upto 127 arguments
  - At a function declaration, you do not need to give a name for each argument

Local Variables

- Automatically, a local variable is created as a function is called, and reclaimed as the function terminates
  - you can denote a local variable with an attribute `auto`

- A function parameter is also a local variable whose initial value is passed into the function

- If `static` is placed in front of a local variable declaration, the variable is created when the program starts and not removed when a function call is terminated
  - a static variable is initialized as zero if no specific initial value is given
  - once a static variable is defined, it will have the same value in the next function call

Global Variable

- A global variable belongs to a program, not to a specific function
  - a global variable is declared at a global scope outside of any function
  - a global variable can be referenced at any part of the program

- A global variable is static
  - it is created when the program is executed
  - the default initial value is zero

- Ex. `convert.c`

The `main` function

- The main function is where the program begins executing
  - no need to write a declaration

- The return value indicates the program's exit status

- The main function may accept parameters from the command line
  - a command-line argument is a string that the user gives for running the program
  - Ex. `command.c`