

## Some C Programming Concepts

I. C programs can be written using any text editor (eg. TextEdit on a Mac, Notepad on a PC) but unlike *interpreted* programming languages (in which programming instructions can be interpreted and executed directly from text format), C programs must first be *compiled* into a binary file in order to run. So in addition to a text editor one must also have a C compiler in order to run C programs. The programs that you write using a text editor (human-readable, with comments, etc.) are referred to as *source code*. As a convenience, and sometimes for profit) there are often *Integrated Development Environments* (IDEs) available which allow a programmer to edit code and compile it from the same interface. The Machine-Science interface functions as an IDE.

### II. Compilation of C programs happens in 3 stages:

#### 1. Preprocessing

In this stage, all comments in the code are discarded (they are there for humans to read, not machines.) Then any preprocessor directives (like including external code from “#include” statements) are performed. This changes the program in memory to include the extra code, but doesn’t change the source code saved to disk.

#### 2. Compiling

This process creates a binary file called “object code” from your source code.

#### 3. Linking, building executable

In the final stage a part of the compiler called a “Linker” links the object code resulting from your “main” code with any other object code necessary to build an executable program. In the case of a desktop application, this would be something you could double-click on and it would run. In the case of micro-controllers, the result is called *bytecode*. This is specific to each chip architecture (eg. Atmega168) and requires a final step of being downloaded to the chip.

### III. Primary Data Types

All C compilers accept the following basic data types:

<u>data type</u>	<u>name</u>	<u>range of values</u>
void	void	empty (non-zero) value
Character	char	-128 to 127
Integer	int	-32768 to 32767
Long Integer	long	-2147483648 to 2147483647
Floating Point	float	$3.4 * -10^{38}$ to $3.4 * 10^{38}$
Double Precision Float	double	$1.7 * -10^{308}$ to $1.7 * 10^{308}$

### IV. Using Variables

Some things to remember about variable **names**:

- They are case-sensitive (eg. data, Data and DATA are considered 3 different variables.)
- They must begin with a letter. the rest must be letters, numbers or underscores
- Variables in a program whose values don’t change are referred to as *constants*. These are traditionally written in all caps so you can recognize them as such. (eg. MAX\_WIDTH).

### V. 3 Main Types of Operators

In order to respond dynamically to changes in the system as the program is running (eg. responding to a user closing a switch, etc.) C programs can perform operations on data. The 3 main ones are:

#### 1. Assignment

`int num = 4` assigns a value of 4 to a variable “num” (*Note: type must be specified first.*)

#### 2. Mathematical operations

addition	subtraction	multiplication	division	modulo	increment	decrement
+	-	*	/	MOD	++	--

#### 3. Boolean Operations

greater than	less than	greater than or equal	less than or equal	equal to	not equal to
>	<	>=	<=	==	!=