Starting off in C

Friday, February 15, 2013 8:43 PM

Quick History: C is a widely used programming language invented by Dennis Ritchie in Bell Laboratories and laid the ground work for many language such as C++ and Java.

The below program is a famous "hello world" program that demonstrates many things about the C language.

```
1  #include <stdio.h>
2
3  // This is an oldie but Goodie
4  int main() {
5
6      printf("hello world");
7      /**
8       * The above statement
9       * prints out hello  world.
10      */
11
12
13     return 0;
14
15 }
```

What is a String?
A string is a message that is in words
And contained by quotation marks such
As "hello world" almost any character can
Be put in the quotation marks such as
"7 is a $money number". A string is used
to represent words mainly.

```
#include <stdio.h>
```

The statement on line 1 "#include <stdio.h>" Tells something called the preprocessor to link the contents of this file in the program. For now think of this statement as a way of importing extra functionality in our program That has been made for us in the C library.

```
// This is an oldie but Goodie
```

The statement on line 3 is a **one line comment** and is used to tell anyone looking at the code more about it. It is important to comment your code so others and yourself and understand what is going on when looking back at it.

```
int main() {
```

The declaration on line 4 is ESSENTIAL to every C program. This is a function declaration. In programming functions are used to interact and change data based on the design of the program. The "int" is short for integer which is a whole number value (-2,0,1,1,000,000, etc..) the "int" before the function name is saying that an int Will be returned inside the braces of this function (line 12)I will make sure to go into detail about **return** later. Just know that a function can return nothing or a type of something (such as an integer) for the program to use.

The "main" is the name of the function and every c program must have a program named main that returns an int, this is what the **compiler** looks for. The compiler is what transforms the human code into 1's and 0's for the computer to understand, it is a wonderful thing. When you run a C program, whatever is in the main program will be run automatically. It is important to add that ONLY the main function will be run as well.

The "()" braces after main indicate the number of parameters the function takes.

```
int main(int i) {
```

Now I have a parameter of the integer type we just learned about! Notice I gave the parameter a name "(i) this is so I can easily access it. Think of it as when you have a variable x in math, you know x refers to a number but you refer to it as the variable x. This is the same in the C language. Parameters are important because sometime you need to know information, to do certain things. For instance, you can not just "add" you need two numbers to add together. Those two numbers can be thought as parameters needed to complete the function of addition.

Now after the braces there is a bracket "{" this shows that the BODY of the function is beginning. Every brace that is opened must be CLOSED (the opening brace on line 4 is closed on line 14). The same goes for parentheses.

```
printf("hello world");
```

Line 6 shows the **printf** function, this function (which is imported from the C library we call to on line 1) simply displays a printed value on the screen. This function takes a parameter of a **string**.

The semicolon at the end(;) notes that the this is the end of the statement. ALL statements must end with a semicolon, this can be a common error in programming and get used to using a semicolon, you will see it a lot more. Notices how the parentheses surround the string. This tells the function where the parameters begin and end.

```
/**
 * The above statement
 * prints out hello  world.
 */
```

Lines 7-10 is a **Multi-Line Comment.** This is used to make a comment that would be longer than one line. All multi-line comments must start with "/**" and end with "*/" to work right. Comments will in NO way effect your code.

Line 13 is the return statement. As I said before the int before main on line 4 indicates that this function must return an "int" and so to return a value we simple type "return 0;" for this function of main. We could return 100 or a variable as long as it has an integer type. Please Note that any code written after the return statement will NOT be executed when the return statement is reached in code. That is saying

that the function is done and to return the indicated value.

As we will learn later function can return different things then "int" or nothing at all as well.

Then line 15 has a closing brace which closes the beginning brace on line 4 The closing brace ends the main function and is required to show the beginning and end of all functions..

Essential Things to remember

- An "int" stands for integer which is a whole number.
- A function called "main" with a return type of int must be in EVERY C program.
- Functions that have as little as 0 or as many parameters as they need/ want. But they must be separated by a comma(see example below)

```
int main(int big, int small, int medium)
```

- To include a C library in your function you must use a "#include" followed by "<" and ending with a ">" to tell the program that you are looking to include already existing functions from the C library.
- There are special keywords in C such as "int" and "return" that are reserved words and mean special things in the C language. Each of these keywords can do very special things and each for a different situation. This is why the keywords are purple in the examples above and non-keywords are black.