Introduction to C

CMSC 104 Section 2
February 19, 2024

Administrative Notes

Here’s a C program - explanation on next slide

```
// This is a comment
// This program prints welcome to c on the screen
#include <stdio.h>
int main()

{

  printf("Welcome to c programming\n");
  printf("Welcome to C ");
  printf("programming \n\n");
  printf("This is output");

    }
```

Anything that starts with a # is a command to the *pre-processor*

Here's a C program

```
// This is a comment
// This program prints welcome to c on the screen
#include <stdio.h>
int main()

{

  printf("Welcome to c programming\n")
    printf("Welcome to C ")
    printf("programming \n\n")
    printf("This is output")

    }
```

Comments are notes to the programmer and other humans about what the program is trying to do and why

\n is an escape code that means "print a newline"

This is a function, or function block, or block, of code. A function is a group of code that does something. There has to be a function called "main" for it to be a valid C program

Some more notes on that program

- Anything between two sets of double quotes is a ***literal***
  - It means “exactly this and nothing else”
  - The strings in the print statements will be printed exactly as you see them
    - Keeping in mind the \n escape codes
- The printf function stops printing in the next column after you’re done
  - If you don’t specify a new line, you won’t get a newline
  - Keep that in mind when you’re formatting your output so that a human can read it
-

Now a slightly more complicated program

```
//This is a c program that prompts the user for two integers,
/* reads them in,
   adds them,
   and then prints out the result
*/

#include <stdio.h>
int main ()
{
  int first_integer;  // the first integer the user will type in
  int second_integer; // the second integer the user will type in
  printf("This program will read in two integers, add them and print the result\n");

  printf("Please enter the first integer \n");  //prompt for the first integer
  scanf("%d", &first_integer);  // read the first integer

  printf("Please enter the second integer \n"); //prompt for the second integer
  scanf("%d", &second_integer); //read the second integer

  int sum;   //the variable where we will store the sum
  sum = first_integer + second_integer;

  printf("The sum of your numbers is %d\n", sum); //print the sum
}
```

This is a block comment. Everything between /* and */ is part of a single comment

Here we are declaring variables. We're telling the C compiler we're going to use these identifiers in the program, and how. "Int" means they will hold decimal integer numbers

scanf() is a function that reads in some value from the keyboard. "%d" means it will be a decimal integer number. What happens if the user doesn't type an integer?

This is standard integer math. Add two numbers together. It works the same way you'd expect.

Now print out your result. Again, %d means a decimal integer.

Some common escape codes in C:

Code	Meaning
\n	Newline
\t	Tab
\\	Insert an backslash character
\”	Insert a double quote character.