EECS10 Discussion Week1

TA: Emad Arasteh
emalekza@uci.edu
eecs10@eecs.uci.edu

Office Hours: Fri, 8:00-9:00am EH 3404

University of California, Irvine

UCIrvine
University of California, Irvine

A simple C program

- Program Comments
  - Start with /* end with */
  - Be ignored by the compiler
  - Should be use to enhance readability

```
/* HelloWorld.c: our first C program    */
/*                                      */
/* author: Rainer Doemer                */
/*                                      */
/* modifications:                       */
/* 09/28/04 RD  initial version         */

#include <stdio.h>

/* main function */

int main(void)
{
    printf("Hello World!\n");
    return 0;
}

/* EOF */
```

- Preprocessor directive #include
  - Insert a header file into the source code

- Standard header file
  - Comes from the C standard library
  - Contains the declarations of standard types and functions for data input and output (e.g. printf(), scanf())

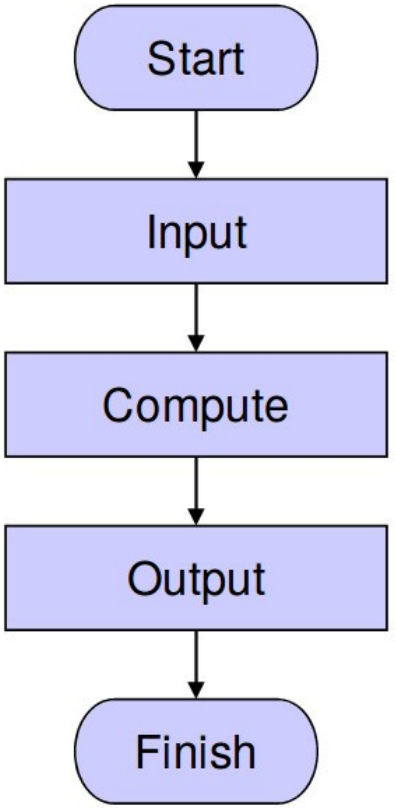
Parts of simple C program

- int main (void)
  - Main function of the C program
  - Program entry (execution starts and ends)
  - Must return an integer (int) value to the operating system at the end of its execution
    - Return 0 indicates successful completion
    - Return value greater than 0 usually indicates an error condition
- Function body
  - Block of code (definitions and statements)
  - Starts with an opening curly brace ({)
  - Ends with a closing curly brace (})
- Printf() function
  - Formatted output (to stdout)
- Return statement
  - Ends a function and returns its arguments as result

```
/* HelloWorld.c: our first C program    */
/*                                      */
/* author: Rainer Doemer                */
/*                                      */
/* modifications:                       */
/* 09/28/04 RD  initial version         */

#include <stdio.h>

/* main function */

int main(void)
{
    printf("Hello World!\n");
    return 0;
}

/* EOF */
```

Assignment 2 (Part 1)

- Program structure
  - Input
    - read input data: scanf()
  - Compute
    - Compute output data from input data
  - Output
    - Write output data



Assignment 2 (Part 1)

- Program structure
  - Initialization section
    - Variable definition
  - Input section
    - Standard input function: scanf()
  - Computation section
    - Perform the necessary computation on variables
  - Output section
    - Standard output function: printf()
  - Exit section
    - clean up and exit: exit 0

- Let’s do some coding by calculating the volume of a cube.

Assignment 2 (Part 2)

- Similar to Part 1
  - Input
    - Standard input function: scanf()
  - Compute
    - Use arithmetic operations: +, -, *, /, %, <<, >>, ...
  - Output section
    - Standard output function: printf()

- Carry over in addition
- Dividend = divisor * quotient + remainder

- Let’s do some coding by adding two timestamps:
  - Timestamp1: 43 seconds
  - Timestamp2: 20 seconds
  - Addition: 63 seconds ?
  - 1 minutes, 3 seconds