

# A Swim in the C

CMS 230, Fall 2017

**Due Thursday, September 7, 11:59:59 PM**

## Description

The following problems will give you practice writing basic programs in C, as well as reviewing some fundamental programming concepts like loops and recursion.

This assignment is not supposed to be difficult—it's mainly intended to be practice for using the terminal environment, the compiler, and GitHub. See me quickly if you find yourself struggling with any of these problems.

## Reading

In addition to these questions, please complete all questions for chapter 5 of the course ZyBook by the project deadline.

## Files and Grading

Submit your code to your own **Project-1** GitHub repository.

Submit one `.c` source file for each problem. Your files must be named `problem1.c`, `problem2.c`, etc.

Include a **Makefile** that builds your executables from source when `make` is run at the command prompt. The executables must be named `problem1`, `problem2`, and so forth.

I will use the included Python script `test.py` to grade your code. To run it, type

```
prompt$ python test.py
```

The test script will build your code using `make`, then run your five programs. Each program's output is compared to the expected output given in the `.cmp`

files located in the `tests` directory. You will pass a test if there are **no differences** between your output and the test's expected output.

Your grade will be the percentage of tests you pass. A working build counts as one of the tests. I will also manually examine your files to make sure they have reasonable style and employ good problem solving strategies (i.e. you can't just print the expected answers). I reserve the right to deduct points for poor programming style.

## The Problems

### OHAI DERE

Write a program that prints `Hello, World!`.

### Super Mario Brothers Super Show

At the end of most levels of the original *Super Mario Brothers*, Mario jumps up a staircase like this:

```
##
###
####
#####
#####
#####
#####
#####
#####
```

Write a C program that can produce such a staircase, with the height controlled by a variable. Print a staircase that is 8 steps high.

You must use loops to print the correct staircase for any positive value of the height variable—don't just manually print the answer for 8 steps!

*Hint:* if the height is  $h$ , the top level has  $h-1$  spaces followed by two `#` characters. The next row has  $h-2$  spaces followed by three `#` characters, and so forth.

### FizzBuzz

A famous programming interview question described by Imran Gohry. Write a program that loops through the numbers from 1 to 30. Print each number, except

- For numbers divisible by 3 print `Fizz`

- For numbers divisible by 5 print **Buzz**
- For numbers divisible by 3 and 5 print **FizzBuzz**

The output for the the first six numbers will look like this:

```
1
2
Fizz
4
Buzz
Fizz
```

### Everyone Needs a Hobby

I enjoy building stone ziggurats in my backyard. To build an  $N$ -level ziggurat, I first build an  $N \times N$  square of stones on the ground. Then I build an  $N-1 \times N-1$  square of stones for the second level, then an  $N-2 \times N-2$  square of stones for the third level, and so forth, until I finally place a single stone on the top level.

Write a **recursive** C program that calculates the number of stones in a ten-level ziggurat. Your program **must use a recursive function** called **stones** to perform the calculation.

*Hint:* The number of stones in a ten-level ziggurat is the number in a nine-level ziggurat plus  $10^2$ . In general,

$$\text{stones}(N) = \text{stones}(N-1) + N^2$$

### Binet's Formula and Linking with Libraries

Recall the famous Fibonacci sequence, where each term is the sum of the two previous terms:

$$1, 1, 2, 3, 5, 8, 13, 21, 34, \dots$$

Binet's Formula (named after the mathematician Jacques Philippe Marie Binet) is an explicit formula for finding terms in the Fibonacci sequence. The  $n$ th Fibonacci number,  $F_n$ , is given by

$$F_n = \frac{1}{\sqrt{5}} \left( \left( \frac{1 + \sqrt{5}}{2} \right)^n - \left( \frac{1 - \sqrt{5}}{2} \right)^n \right)$$

The special number

$$\phi = \frac{1 + \sqrt{5}}{2} \approx 1.618033\dots$$

is the famous *golden ratio*, the most aesthetically pleasing of all proportions.

The formula can be derived using **generating functions**, which are awesome but not part of this course.

Write a C program that uses Binet's Formula to calculate and print the first 10 Fibonacci numbers. Use `sqrt` and `pow` to perform the calculations; both functions are defined in `math.h`. Look up both commands to see how they're used.

To use the `pow` function, you'll need to **link** your code with the math library. **Libraries** are pre-compiled collections of useful routines. The linking process merges this pre-compiled code into your executable.

By convention, all libraries start with the prefix `lib-`, followed by the name of the library. The math library is called `libm.a` and lives in a subdirectory of `/usr/lib`.

Use the `-l` flag to link a library. The appropriate command is

```
gcc -Wall -Werror -o problem5 problem5.c -lm
```

`gcc` processes the `-l` flag by interpreting the rest of the flag—the letter `m`—as the name of a library. It uses that name to generate a library name in the standard form (`libm.a`), then looks up the library and adds it to your program.

## Handing It In

You will submit your project as a GitHub repository.

Here's the basic procedure you'll follow. We'll go over this in class; you'll have a chance to practice working with GitHub before the assignment is due.

- I have created a GitHub group called `rollinscms230`. This is the entry point for all your project submissions.
- When I create an assignment for the group, GitHub automatically generates a link that I will send to you.
- Clicking on that link creates a private repository where you will store your code for the project. Only you and I have access to the repo.
- As you work on the assignment, push your updates from Cloud9 to your GitHub repo.
- When it's time to grade the projects, I'll pull your code from your repo and run the test script against your files.