

## Fizzbuzz

Write a program that prints the numbers from 1 to 100. But for multiples of three print "Fizz" instead of the number and for the multiples of five print "Buzz". For numbers which are multiples of both three and five print "FizzBuzz".

## Gettysburg

You'll find a text file called `Gettysburg.txt` on the desktop. You'll be writing a program which will format the text so that each line will be at most 13 characters per line, breaking the line at the appropriate spaces. For extra credit, break apart words with greater than 13 characters into two words separated by an hyphen. An example of the expected output of your program can be found in `Gettysburg.ouput`.

Before starting this task provide three time estimates, a low bound, a high bound, and your best guess for how long it will actually take for you to write this program. Keep in mind that this calculation isn't difficult for a human to perform manually - if you spent one second per word determining whether or not that word is the appropriate break point for a line, it would take you just under five minutes to break the entire address up into lines that are 13 characters or less.

## McIlroy

Discuss how you would go about writing a program or script to read a file of text, determine the  $n$  most frequently used words, and print out a sorted list, from high to low, of those words along with their frequencies, e.g:

```
10 foo
9 bar
6 foobar
```

Then write the program using the Gettysburg.txt file on the desktop for example input (expected output can be found in Gettysburg.output2).

## Phone numbers

I have a directory containing thousands of plain text files (some in subdirectories). I want to find all of the files containing a phone number, in one of the following two formats:, and copy them into a destination directory. You should produce an executable, which can be invoked as `find-phone-numbers <source-dir> <target-dir>.v`