

Software Developer Test Question

The goal is to implement a program that counts the lines of codes/comments/blanks in a C/C++ project, similar to the `cloc` command:

```
$> cloc my-project-dir
```

| Files | Lines | Code | Comments | Blanks |
|-------|-------|------|----------|--------|
| 32 | 7153 | 6228 | 442 | 483 |

Attention: Lines = Code + Comments + Blanks.

Please search for C/C++ language syntax and be aware of edge cases, e.g. comment lines in String, etc.

Specification

The command line accepts one parameter which is a directory of source code. It needs to traverse subdirectories to count all files with extension c, cpp, cc.

You can assume that all source files follow the standard C++11 grammar.

Please respect the following points:

- Only count files with relevant extensions only, e.g. do not include h (head file).
- The output format must follow the `cloc` output format.
- Take care of edge cases.

The C++ project called 'Osiris 2' is present in the drive that has been shared to you.

Test example

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

```
/*
```

```
    This is a multi-line comment
```

```
    The blanks in multi-line comment are counted as Blanks
```

```

*/

int main() {

    /// This is a comment line
    printf ("Hello, world!"); // Code+Comment line together counted as Code.

    return 0;

}

```

Output as:

| Files | Lines | Code | Comments | Blanks |
|-------|-------|------|----------|--------|
| 1 | 12 | 5 | 5 | 2 |

Requirement:

- Run the code to count lines of files inside of the project 'Osiris 2', and copy the output result to email.
- Attach your solution code with documentation on how to compile the project.
- Content should not include any binary files.
- Good documentation is a plus
- Good code readability is a plus

Extra points

Add support to Ruby language if you have the time.

The Ruby project called 'lib 2' is present in the drive that has been shared to you.

Test example

```
=begin
```

```
This is multi-line comment
```

```
=end
```

```
## Single line comment
```

```
puts "Hello, world!"  Comment line within # counted as Code Line.
```

Output as:

| Files | Lines | Code | Comments | Blanks |
|-------|-------|------|----------|--------|
| 1 | 7 | 1 | 4 | 2 |

Requirements:

- Run the code to count lines of files of the project 'lib 2', copy the output result to email.
- Ruby and C++ implementations must be in the same project: you can design a command line option to specify the language being counted and provide documentation on how to run your program.
- Counting lines of codes of a project using a mix of both languages is not required.