

CSE 311: Foundations of Computing I

OnMySourceCodeGENERATOR

Notation

We write $\text{CODE}(\mathbf{P})$ to mean “the source code of the program \mathbf{P} ”. We also write $\mathbf{P}(x, y, z, \dots)$ to mean the “the output of the program \mathbf{P} when given x, y, z, \dots as input”; Note that by output, we mean all the `System.out.print` and `System.out.println` statements (not the return value).

Quines and Identities

- A quine is a program \mathbf{Q} such that $\mathbf{Q}() = \text{CODE}(\mathbf{Q})$. That is, it is a program that outputs its own source code when given no input.
- An identity program is a program \mathbf{Id} such that $\mathbf{Id}(x) = x$. That is, it is a program that outputs exactly its input.

$\mathbf{P}\text{OnMySourceCode}$

Let \mathbf{P} be some particular static method that takes some string as an argument and outputs a string.

$\mathbf{P}\text{OnMySourceCode}$ is a program that takes no arguments such that

$$\mathbf{P}\text{OnMySourceCode}() = \mathbf{P}(\text{CODE}(\mathbf{P}\text{OnMySourceCode})).$$

Here are some examples of $\mathbf{P}\text{OnMySourceCode}$ for various programs \mathbf{P} :

- If $\mathbf{P} = \text{Length}(x)$, then $\text{Length}\text{OnMySourceCode}()$ should print the length of its source code.
- If $\mathbf{P} = \text{AllUpperCase}(x)$, then $\text{AllUpperCase}\text{OnMySourceCode}()$ should print its source code in all upper case letters.

Note that if $\mathbf{P} = \mathbf{Id}$, then $\mathbf{Id}\text{OnMySourceCode}$ is just a quine!

OnMySourceCodeGENERATOR

Your task for this question is to write the program `OnMySourceCodeGENERATOR`.

`OnMySourceCodeGENERATOR` is a program that given the source code to some particular program \mathbf{P} , outputs the source code to the program $\mathbf{P}\text{OnMySourceCode}$.

You should name the class that you output to $\mathbf{P}\text{OnMySourceCode}$, and since it should take no input, *main* should execute \mathbf{P} on $\mathbf{P}\text{OnMySourceCode}$'s source code.

In other words, you will be writing a program that constructs and prints $\mathbf{P}\text{OnMySourceCode}$ when given $\text{CODE}(\mathbf{P})$ as input. Intuitively, we are asking you to write this program instead of directly writing $\mathbf{P}\text{OnMySourceCode}$, because it's easier to let the computer generate $\mathbf{P}\text{OnMySourceCode}$ rather than doing it directly.

Writing Your Program

You are given several Java classes which we explain here:

`OnMySourceCodeGENERATOR.java` This is the file you will be editing. You should not edit `main`, and you should not edit any other files. This file should write out a class, `POnMySourceCode`, where **P** is the code input to `OnMySourceCodeGENERATOR` using `System.out.println`.

`Tester.java` This program will test your implementation of `OnMySourceCodeGENERATOR` using `P.input` as the input to your `OnMySourceCodeGENERATOR` function.

`P.input` This file will contain a single (valid) static Java method called **P** which takes exactly one `String` argument.

You are permitted, and even encouraged, to add helper functions to the source files to handle repetitive tasks. Output everything using `System.out.print` and `System.out.println`.

You may not do any disk or network I/O of any kind on this assignment, nor may you use things like the `system` function.



You should only need to use methods from the following Java classes:

- `System.out`
- `String`

Tips!

Long Java string constants: If you have a very long string constant in Java, you can make it out of smaller strings concatenated with a plus sign. Note that in Java you can concatenate ints and chars with string very easily:

```
str = "This is a very long Java string "+
      "constant. It spans multiple "+
      "lines in the program, but when "+
      "the program is run, this "+
      "will just be one long string "+
      "with " + 1 + " newline at the end.\n";
```

String Escape Sequences: One way of handling this assignment involves lots of escape sequences (like `\n` `\t` `\\` `\"`), but there are better ways to do it. In particular, the fact that characters are numbers can be used to deal with this. There is a really cool reading which is both interesting and helpful for this problem; the part most relevant to this assignment is the part about the quine (see Figure 1).