2019 MCS High School Workshop

CS Fundamentals and First-Year Assessments

Michael Liut & Paul Vrbik

michael.liut@utoronto.ca

paul.vrbik@utoronto.ca

University of Toronto Mississauga

May 1, 2019

CS 108 (Introduction to Programming)

Question (Assignment 0)

This assignment is not meant to evaluate your ability to program or your comprehension of the program thus far. The assignment is designed to get you comfortable with submitting your assignments electronically, and expose any issues which may arise.

Implement (i.e. complete the code of) a single function

```
calc_volume(float, float, float) -> float
```

which returns the volume of a rectangular prism given the length, width, and height (respectively).

Outcome

Out of 1100 students. . .

Academic offences	40
Zero grade	200

What happened?

1. Introducing syntax errors in the starter code.
   - Copying from IDE and included `>>>` in cases.
   - Insufficient understanding of `file.py`.
2. Not uploading properly (inability to follow instructions).
3. Renaming files and definitions which autogrades to zero.

   Amazingly the students cannot fathom how or why we use programming to automate a routine task.
4. Using `print` rather than `return`.
   - It took nearly all semester to train this out of them.
   - Worth looking at more closely …

```
>>> def foo(x):
        return x**x

>>> def bar(x):
        print(x**x)

>>> foo(1)
1
>>> bar(1)
1

>>> x = foo(1)
                    #This is why they print
>>> y = bar(1)
1
>>> x - y
```

Question

What are some effective ways of breaking this habit?

Question

What are some effective ways of breaking this habit?

Answer

Assigning zeros is not effective as this habit persisted well beyond the first assignment.

Some were convinced by seeing the following would break:

1. `x = bar()`,
2. `bar(foo(1))`,
3. `foo(1) - bar(1)`.

Answer (cont...)

The most effective argument was timing the functions.

```
>>> from timeit import timeit

>>> timeit( lambda : foo(2*123456), number=1)
0.2923482660003174

>>> timeit( lambda : bar(2*123456), number=1)
...5464332281367415337045282836944549276209658513363
790473480585551130121384068889925856464045820870656
22.025573137000038

>>> 22.025573137000038 / 0.2923482660003174
75.34018736740558
```

Functions

Question

What is the purpose of functions?

Functions

Question

What is the purpose of functions?

Answer

1. Breaking down large blocks of code into a more modular design:

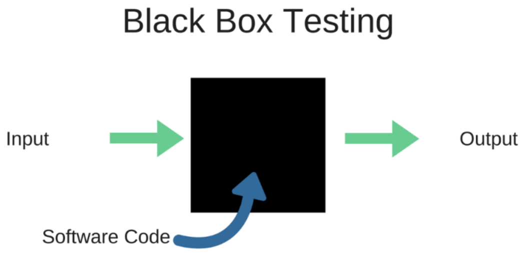
   Low coupling. High Cohesion.

2. Reusability.

3. Abstraction: Naming, Parameters, and What it does.

"You don't need to know how every function works, just how to use it and what it does."

“You don’t need to know how every function works, just how to use it and what it does.”



Suggested Assignment

Question

Given two functions `foo(int) -> int` and `bar(int) -> int` (either globally scoped or passed as arguments) write the function

```
fcomp( ) -> int
```

which returns `foo` $\circ$ `bar`.

Answer

```
>>> #Assume foo and bar are globally scoped
>>> def comp(x):
        return foo(bar(x))

>>> def comp(x):
        tmp = bar(x)
        return foo(tmp)
```