Table of Contents

Sentimental

tl;dr

1. Port `hello.c` to `hello.py`.
2. Port `mario.c` to `mario.py`.
3. Port `cash.c` to `cash.py` or `credit.c` to `credit.py`.
4. Port `caesar.c` to `caesar.py`, `vigenere.c` to `vigenere.py`, or `crack.c` to `crack.py`.

Problems

1. Implement the below exactly as specified but in Python:
    - Hello (../../../../../problems/hello/hello.html), in `pset6/hello/hello.py`
2. Implement either of the below exactly as specified but in Python:
    - Mario (../../../../../problems/mario/less/mario.html), less comfortable, in `pset6/mario/mario.py`
    - Mario (../../../../../problems/mario/more/mario.html), more comfortable, in `pset6/mario/mario.py`
3. Implement either of the below exactly as specified but in Python:
    - Cash (../../../../../problems/cash/cash.html), less comfortable, in `pset6/cash/cash.py`
    - Credit (../../../../../problems/credit/credit.html), more comfortable, in `pset6/credit/credit.py`
4. Implement any (one) of the below exactly as specified but in Python:

- Caesar (../../2/caesar/caesar.html), less comfortable, in `pset6/caesar/caesar.py`
- Vigenère (../../2/vigenere/vigenere.html), less comfortable, in `pset6/vigenere/vigenere.py`
- Crack (../../2/crack/crack.html), more comfortable, in `pset6/crack/crack.py`

Walkthroughs

Note that, among these walkthroughs, Cash was previously called Greedy. Also, for Crack, passwords can be up to 5 chars long (not 4 as shown).

Testing

Correctness

```
check50 cs50/2018/x/sentimental/hello
check50 cs50/2018/x/sentimental/mario/less
check50 cs50/2018/x/sentimental/mario/more
check50 cs50/2018/x/sentimental/cash
check50 cs50/2018/x/sentimental/credit
check50 cs50/2018/x/sentimental/caesar
check50 cs50/2018/x/sentimental/vigenere
```

As before, afraid there's no `check50` for Crack, lest it spoil the fun!

Style

```
style50 hello.py
style50 mario.py
style50 cash.py
style50 credit.py
style50 caesar.py
style50 vigenere.py
style50 crack.py
```

Hints

- Be sure to use Python 3, not Python 2. The former is installed by default on CS50 IDE, but if Google leads you to Python's official documentation, be sure the URLs begin with https://docs.python.org/3/ (https://docs.python.org/3/), not https://docs.python.org/2/ (https://docs.python.org/2/).
- If a program is in a file called, say, `foo.py`, you can run that program with `python foo.py`.
- For Hello, Mario, Greedy, Credit, Caesar, Vigenère, and Crack, it is **reasonable** to look at your own implementations thereof in C and others' implementations thereof in C, including the staff's implementations thereof (and postmortems) in C. It is **not reasonable** to look at others' implementations of the same in Python.
- Consider this problem set an opportunity not only to port your own prior work from C to Python but to improve upon your earlier designs using lessons learned since!
- When porting code from C to Python in CS50 IDE, you might want to select **View > Layout > Horizontal Split** so that you can see both side by side.
- Insofar as a goal of these problems is to teach you how to teach yourself a new language, keep in mind that these acts are not only **reasonable**, per the syllabus, but encouraged toward that end:

- Incorporating a few lines of code that you find online or elsewhere into your own code, provided that those lines are not themselves solutions to assigned problems and that you cite the lines' origins.
- Turning to the web or elsewhere for instruction beyond the course's own, for references, and for solutions to technical difficulties, but not for outright solutions to problem set's problems or your own final project.

- You're welcome to use the CS50 Library for Python, which includes `get_float`, `get_int`, and `get_string`. Just remember to include any of

```
from cs50 import get_float
from cs50 import get_int
from cs50 import get_string
```

  atop your code. Or you can use `input` (https://docs.python.org/3/library/functions.html#input) and validate users' input yourself.
- You might find `chr` (https://docs.python.org/3/library/functions.html#chr) and/or `ord` (https://docs.python.org/3/library/functions.html#ord) of help.
- You might find these references of interest:
  - The Python Language Reference (https://docs.python.org/3/reference/index.html)
  - The Python Standard Library (https://docs.python.org/3/library/)
  - The Python Tutorial (https://docs.python.org/3/tutorial/index.html)