CS 100 Programming I
Project 0

Revision Date: January 8, 2017

Preamble

You may develop your code anywhere, but you must ensure it runs correctly on a Linux distribution before submission.

Payback!

> *A penny saved is a penny earned* – Benjamin Franklin

Dr. Lusth has gotten into hypermiling, where the game is to achieve significantly better gas mileage compared to EPA estimates. One way of doing this is to use certain gas-saving driving techniques such as Pulse and Glide and Driving with Load. Another way is to modify the vehicle so that it becomes more efficient. For example, Dr. Lusth installed a high-flow air filter and drilled holes in his vehicle's air box to increase the amount of oxygen going into the engine cylinders. He figures this modification gained him a 3% increase in his fuel efficiency. He paid $45 for the filter, but doesn't know if the increase in fuel economy was worth the cost of the filter.

Your task is to determine:

- the yearly cost of gasoline without the modification
- the yearly cost of gasoline with the modification
- how long it takes, in years and days, for his investment to pay off

Program Organization

Create a *project0* directory off of your *cs100* directory and move into *project0*.

Name your main python file *payoff.py*. Provide one more file named *mpg.py* The main function in the main file should prompt the user of your program for the following information:

- number of miles driven per year
- current mpg
- cost (in dollars) of a gallon of gasoline

- cost of the modification
- percent increase in mpg after the modification

in the order given.

Your main function should call at least four different functions; one should read in and return the five pieces of information and each of the other should calculate and return the results to be displayed (cost with, cost without, pay off time). The definitions of these four functions should be placed in the *mpg.py* file. You will need to import these functions into your main program by placing the line:

```
from mpg import *
```

at the top of *payoff.py*.

Of course, your *mpg.py* file may have other functions in it, as well.

NOTE: Only your main function should print the results; the functions in *mpg.py* should only compute the values to be displayed, not display them:

```
...
def main():
    # read in the information
    ...
    # find costs
    costWithout = getCostWithout(...)
    print("yearly cost without the modification: ",costWithout)
    ...
    # find payoff time
    ...

main()
```

Getting the numbers

Here is how to get two numbers and return them:

```
def getInfo():
    m = eval(input("Give me a number: "))
    n = eval(input("Give me another number: "))
    return m,n
```

Here is how to use the function *getInfo*:

```
x,y = getInfo()
print("the first number is",x)
print("the second number is",y)
```

How do I determine the costs and pay off time?

Use the mailing list to discuss exactly how to do this. DO NOT POST CODE or you will be banned from the mailing list (or worse).

Compliance Instructions

To make sure that you have implemented your program correctly, create a data file named *test0* that contains five numbers. You should then be able to run the following command:

```
cat test0 | python3 payoff.py
```

Using the data in *test0*, the yearly cost of gasoline, without the modification, should be $675.67. With the modification, the yearly gas cost should be $655.99, Finally, the modification should pay for itself in 2 years and 104 days. Don't worry if your costs have more than two decimal places. Adjust the numbers in *test0* until you get the specified results.

This method of running the program is called "piping in the input from a file". When you actually do this, the prompts your program makes for information will all be strung together on a single line. Don't worry about it; it's a natural consequence of the way the program was run.

If your code fails with a runtime error while running this test, then you will receive a zero for this assignment.

Note that your answers do not have to be correct for your program to be graded, only that it not fail. Of course, correct answers will yield a much higher grade.

Note also, that a prerequisite receive full credit, your functions computing results must return numbers, not strings. HINT: have the function that returns the payoff times return the number of years and then have your main function turn that number into years and days. Assume a day is $\frac{1}{365}$ of a year. Truncate the number of days to a whole number.

Challenges

Try figuring out how to get your functions to return numbers like 645.32 rather than 646.325809201928.

Try to get your program to display output like:

```
2 years and 1 day
1 year and 37 days
49 days
```

instead of:

```
2 years and 1 days
1 years and 37 days
0 years and 49 days
```

Stepwise Refinement

Here you can find a detailed strategy for developing your solution to project0.

Submission Instructions

Change to the *project0* directory containing your assignment. Do an *ls* command. You should see something like this:

```
mpg.py   payoff.py   test0
```

Extra files are OK. Submit your program like this:

```
submit cs100 xxxxx project0
```

Remember to replace `xxxxx` with your instructor name.

Note that *project0* ends in a zero, not a capital O.