CPSC 231 – Fall 2012, Assignment 1

Due: *September 28 2012, 4:00PM*

This assignment has two options. You can choose to work on either of them and submit.

1 Option A. Computing Changes

In this option, you will write a Python program that prompts the user to input the before-tax price of an item and the payment from the customer. Your program should then perform two tasks. First, compute and display the after-tax price, assuming a tax rate of 5%. Second, compute the change owed to the customer, and break that down into dollars, quarters, dimes and pennies, with the total number of coins in the breakdown minimized. Display such breakdown information.

Take input

Your program should first prompt the user to input the before-tax price of an item, in the unit of dollars. Your program should then prompt the user to input the payment from the customer, also in the unit of dollars.

Compute and display after-tax price

Your program should compute the after-tax price of the item, assuming a tax rate of 5%. Note, if the after-tax price includes a certain fraction of a penny, ignore that fraction. For example, if the before-tax price is \$3.99, then $3.99 \times 1.05 = 4.1895$, and the after-tax price should be \$4.18.

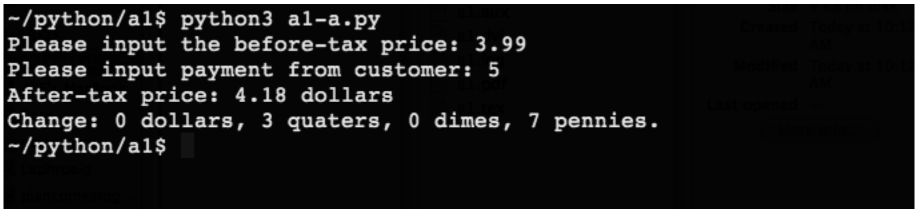
Display this after-tax price, in the unit of dollars.

Compute and display changes

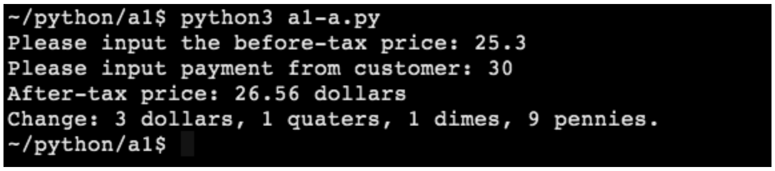
Your program should compute the amount of change owed to the customer, and break this down into dollars, quarters, dimes, and pennies. We assume that 5-cent coins are not used. The program should return larger denominations before smaller ones. For example, if the change is \$1.25, that should be broken down into 1 dollar and 1 quarter, instead of 5 quarters. If the change is \$0.12, that should be broken down into 1 dime and 2 pennies, instead of 12 pennies.

Display this breakdown of changes.

Sample input and output



```
~/python/a1$ python3 a1-a.py
Please input the before-tax price: 3.99
Please input payment from customer: 5
After-tax price: 4.18 dollars
Change: 0 dollars, 3 quaters, 0 dimes, 7 pennies.
~/python/a1$
```

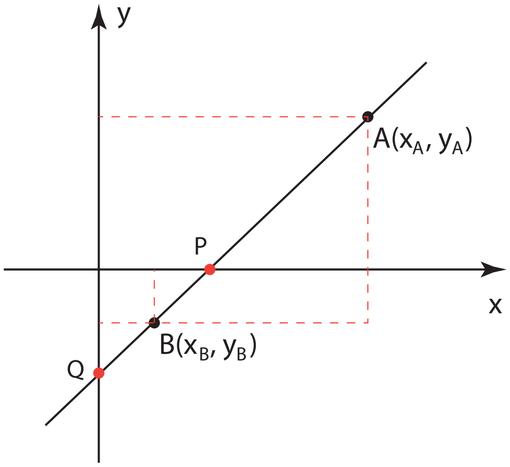


```
~/python/a1$ python3 a1-a.py
Please input the before-tax price: 25.3
Please input payment from customer: 30
After-tax price: 26.56 dollars
Change: 3 dollars, 1 quaters, 1 dimes, 9 pennies.
~/python/a1$ 
```

2 Option B. Lines in 2D

In this option, you will write a program that computes the intersection point of two lines in a 2D coordinate system, as well as the length of a line segment.

Getting input



As shown in the figure above, your program will prompt the user to input four float point numbers, x_A, y_A, x_B, y_B, for specifying the coordinates of two points A and B. You can assume that the line AB is not parallel to either the x-axis or the y-axis; that is, the input will satisfy the conditions that $x_A \neq y_A$ and $x_B \neq y_B$.

Computing Coordinates of P

Let AB be the infinite line that goes through both points A and B. Let $P(x_P, y_P)$ be the intersection point of the line AB and the x-axis. Your program should compute x_P and y_P, and display them.

A line in 2D can be represented using the following equation:

$$y = kx + b$$

If you plug in (x_A, y_A) and (x_B, y_B) to such a line equation, you have:

$$\begin{cases} y_A = kx_A + b \\ y_B = kx_B + b \end{cases}$$

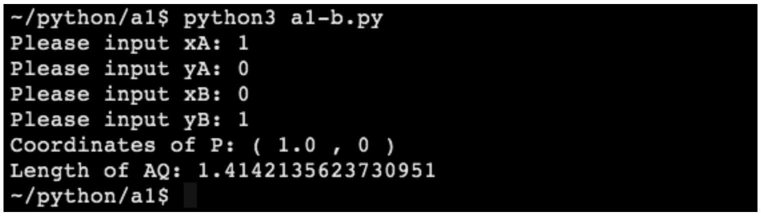
From the above, you can solve for closed form representation of each of k and b, in terms of x_A, y_A, x_B and y_B. From k and b, you should be able to compute x_P and y_P.

Computing The Length of AQ

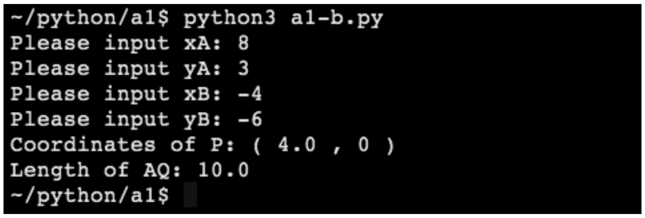
Let Q be the intersection point between AB and the y-axis. Next, your program should compute and display the length of the line segment between A and Q. Your program can first compute the coordinates of Q, (x_Q, y_Q). Then the length can be computed using the following formula:

$$||AQ|| = \sqrt{(x_A - x_Q)^2 + (y_A - y_Q)^2}$$

Sample Input and Output



```
~/python/a1$ python3 a1-b.py
Please input xA: 1
Please input yA: 0
Please input xB: 0
Please input yB: 1
Coordinates of P: ( 1.0 , 0 )
Length of AQ: 1.4142135623730951
~/python/a1$
```



```
~/python/a1$ python3 a1-b.py
Please input xA: 8
Please input yA: 3
Please input xB: -4
Please input yB: -6
Coordinates of P: ( 4.0 , 0 )
Length of AQ: 10.0
~/python/a1$
```

3 Submission

Submit your solution by sending an email to your TA, with your solution program in the attachment. Name your program in the form of `a2-sid.py`. For example, if your student ID is 123456, then name your program `a2-123456.py`.

Submissions received after the deadline will not be accepted.