Change Maker

Submit Assignment

Due Friday by 11pm **Points** 100 **Submitting** a file upload **Available** until Apr 26 at 11pm

Preparation

Online students please watch videos up to lecture 14

Objectives

Create a program that calculates change after a purchase. This will be a *sequential* algorithm, which is one of the simplest programming patterns. Your solution requires no branching or looping. Simply write a series of statements that will execute in order.

Discussion

The program should ask for a purchase price and the amount of cash tendered. It should then determine how many coins or bills of the following denominations should be returned:

penny
$0.01
nickel
$0.05
dime
$0.10
quarter
$0.25
dollar
$1.00
five
$5.00
ten
$10.00
twenty
$20.00

Sample Run

```
Price of the item:
21.37
Cash tendered:
50.00
Change: 28.63
Change Left: 2863
twenties: 1
tens: 0
fives: 1
ones: 3
quarters: 2
dimes: 1
nickles: 0
pennies: 3
```

Notes

This program is easier if you consider the following ideas:

- You'll have to figure out how much of each amount is required
- You'll need to know what's left after you've accounted for each denomination
- Look up the *modulus* operator
- Modulus works better with integers
- It may be better to work in pennies rather than dollars

Submission

Please submit the following on Canvas:

- Your .py file (NOT a link to your pythonanywhere page)
- A .txt file describing your algorithm (congruent with the requirements for algorithm files described in **the announcement (http://%24CANVAS_OBJECT_REFERENCE%24/discussion_topics/https://iu.instructure.com/courses/1784226/discussion_topics/8418801)** about algorithm files)
- If you are turning in a blackbelt version, submit your blackbelt as a separate .py file from your basic .py file as well as a separate .txt algorithm file

Black Belt

There are many ways to extend this. Think about turning it into a vending machine with an inventory, objects with different prices, and different kinds of coin or bill inputs. Or think about the data structure. If you know something about arrays, you might be able to improve this program so it runs cleanly in a loop and you don't have to write the same code over and over. You might also convert a value to another currency and return the appropriate bills and coins in that currency.

As always, be certain to do the main program before adding a black belt version, and turn them in as separate files.