This is CS50x

OpenCourseWare

Donate (https://cs50.harvard.edu/donate)

David J. Malan (https://cs.harvard.edu/malan/)
malan@harvard.edu
(https://www.facebook.com/dmalan) (https://github.com/dmalan)
(https://www.instagram.com/davidjmalan/) (https://www.linkedin.com/in/malan/) (https://orcid.org/0000-0001-5338-2522)
(https://www.quora.com/profile/David-J-Malan) (https://www.reddit.com/user/davidjmalan) (https://www.tiktok.com/@davidjmalan)
(https://twitter.com/davidjmalan)

Credit

Implement a program that determines whether a provided credit card number is valid according to Luhn's algorithm.

```
$ python credit.py
Number: 378282246310005
AMEX
```

Getting Started

Log into code.cs50.io (https://code.cs50.io/), click on your terminal window, and execute `cd` by itself. You should find that your terminal window's prompt resembles the below:

```
$
```

Next execute

```
wget https://cdn.cs50.net/2021/fall/psets/6/sentimental-credit.zip
```

in order to download a ZIP called `sentimental-credit.zip` into your codespace.

Then execute

```
unzip sentimental-credit.zip
```

to create a folder called `sentimental-credit`. You no longer need the ZIP file, so you can execute

```
rm sentimental-credit.zip
```

and respond with "y" followed by Enter at the prompt to remove the ZIP file you downloaded.

Now type

```
cd sentimental-credit
```

followed by Enter to move yourself into (i.e., open) that directory. Your prompt should now resemble the below.

```
sentimental-credit/ $
```

Execute `ls` by itself, and you should see `credit.py`. If you run into any trouble, follow these same steps again and see if you can determine where you went wrong!

Specification

- In `credit.py`, write a program that prompts the user for a credit card number and then reports (via `print`) whether it is a valid American Express, MasterCard, or Visa card number, exactly as you did in Problem Set 1, except that your program this time should be written in Python.
- So that we can automate some tests of your code, we ask that your program's last line of output be `AMEX\n` or `MASTERCARD\n` or `VISA\n` or `INVALID\n`, nothing more, nothing less.
- For simplicity, you may assume that the user's input will be entirely numeric (i.e., devoid of hyphens, as might be printed on an actual card).
- Best to use `get_int` or `get_string` from CS50's library to get users' input, depending on how you to decide to implement this one.

Usage

Your program should behave per the example below.

```
$ python credit.py
Number: 378282246310005
AMEX
```

Hints

- It's possible to use regular expressions to validate user input. You might use Python's `re` (https://docs.python.org/3/library/re.html) module, for example, to check whether the user's input is indeed a sequence of digits of the correct length.

Testing

While `check50` is available for this problem, you're encouraged to first test your code on your own for each of the following.

- Run your program as `python credit.py`, and wait for a prompt for input. Type in `378282246310005` and press enter. Your program should output `AMEX`.
- Run your program as `python credit.py`, and wait for a prompt for input. Type in `371449635398431` and press enter. Your program should output `AMEX`.
- Run your program as `python credit.py`, and wait for a prompt for input. Type in `5555555555554444` and press enter. Your program should output `MASTERCARD`.
- Run your program as `python credit.py`, and wait for a prompt for input. Type in `5105105105105100` and press enter. Your program should output `MASTERCARD`.
- Run your program as `python credit.py`, and wait for a prompt for input. Type in `4111111111111111` and press enter. Your program should output `VISA`.
- Run your program as `python credit.py`, and wait for a prompt for input. Type in `4012888888881881` and press enter. Your program should output `VISA`.
- Run your program as `python credit.py`, and wait for a prompt for input. Type in `1234567890` and press enter. Your program should output `INVALID`.

Execute the below to evaluate the correctness of your code using `check50`. But be sure to compile and test it yourself as well!

```
check50 cs50/problems/2022/x/sentimental/credit
```

Execute the below to evaluate the style of your code using `style50`.

```
style50 credit.py
```

How to Submit

In your terminal, execute the below to submit your work.

```
submit50 cs50/problems/2022/x/sentimental/credit
```