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
```

Specification

- In `credit.py` in `~/pset6/credit/`, 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/2020/fall/sentimental/credit
```

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

```
style50 credit.py
```

This problem will be graded only along the axes of correctness and style.

How to Submit

1. Download your `credit.py` file by control-clicking or right-clicking on the file in CS50 IDE's file browser and choosing **Download**.
2. Go to CS50's Gradescope page (https://www.gradescope.com/courses/157004).
3. Click "Problem Set 6: Sentimental (Credit)".
4. Drag and drop your `credit.py` file to the area that says "Drag & Drop". Be sure it has the correct filename!
5. Click "Upload".

You should see a message that says "Problem Set 6: Sentimental (Credit) submitted successfully!" You won't see a score just yet, but if you see the message then we've received your submission!