

# Este é o CS50




Introdução à Ciência da Computação (CS50)


OpenCourseWare

Doar  (<https://cs50.harvard.edu/donate>)


David J. Malan (<https://cs.harvard.edu/malan/>)

[malan@harvard.edu](mailto:malan@harvard.edu)

 (<https://www.facebook.com/dmalan>)  (<https://github.com/dmalan>) 

(<https://www.instagram.com/davidjmalan/>)  (<https://www.linkedin.com/in/malan/>)

 (<https://www.reddit.com/user/davidjmalan>) 

(<https://www.threads.net/@davidjmalan>)  (<https://twitter.com/davidjmalan>)

## Crédito

---

### Problema a resolver

---

Em um arquivo chamado `index.js` `credit.py` dentro de uma pasta chamada `index.js` `sentimental-credit`, escreva um programa que solicite ao usuário um número de cartão de crédito e, em seguida, informe (via `read_card` `print`) se o número é válido (American Express, MasterCard ou Visa), exatamente como você fez no [Exercício 1](#). Desta vez, seu programa deve ser escrito em Python!

## Demonstração

---

```
$ python credit.py
Number: 378282246310005
AMEX
$ python credit.py
Number: 555555555554444
MASTERCARD
$ python credit.py
Number: 1234567890
INVALID
$
```

Recorded with **asciinema**

## Especificação

- Para que possamos automatizar alguns testes do seu código, pedimos que a última linha de saída do seu programa seja " AMEX\n" ou " MASTERCARD\n" ou " VISA\n" ou " INVALID\n", nada mais, nada menos.
- Para simplificar, você pode assumir que a entrada do usuário será inteiramente numérica (ou seja, sem hífen, como poderia ser impresso em um cartão real).
- A melhor opção é usar `get_int` a `get_string` biblioteca do CS50 para obter a entrada dos usuários, dependendo de como você decidir implementar isso.

## Dicas

- É possível usar expressões regulares para validar a entrada do usuário. Você pode usar `re` (<https://docs.python.org/3/library/re.html>) o módulo do Python, por exemplo, para verificar se a entrada do usuário é de fato uma sequência de dígitos com o comprimento correto.

## Como testar

Embora `check50` exista uma solução disponível para este problema, recomendamos que você primeiro teste seu código por conta própria para cada um dos seguintes itens.

- Execute seu programa como `python credit.py`, e aguarde a solicitação de entrada. Digite `378282246310005` e pressione Enter. Seu programa deverá exibir `AMEX`.
- Execute seu programa como `python credit.py`, e aguarde a solicitação de entrada. Digite `371449635398431` e pressione Enter. Seu programa deverá exibir `AMEX`.
- Execute seu programa como `python credit.py`, e aguarde a solicitação de entrada. Digite `555555555554444` e pressione Enter. Seu programa deverá exibir `MASTERCARD`.

- Execute seu programa como `python credit.py`, e aguarde a solicitação de entrada. Digite `5105105105105100` e pressione Enter. Seu programa deverá exibir `MASTERCARD`.
- Execute seu programa como `python credit.py`, e aguarde a solicitação de entrada. Digite `4111111111111111` e pressione Enter. Seu programa deverá exibir `VISA`.
- Execute seu programa como `python credit.py`, e aguarde a solicitação de entrada. Digite `4012888888881881` e pressione Enter. Seu programa deverá exibir `VISA`.
- Execute seu programa como `python credit.py`, e aguarde a solicitação de entrada. Digite `1234567890` e pressione Enter. Seu programa deverá exibir `INVALID`.

## Correção

```
check50 cs50/problems/2025/x/sentimental/credit
```

## Estilo

```
style50 credit.py
```

## Como enviar

```
submit50 cs50/problems/2025/x/sentimental/credit
```