(/)

You have a captain's log due before 2020-07-19 (in about 3 hours)! Log it now! (/captain_logs/39703/edit)

0x00. Lockboxes

Specializations - Interview Preparation — Algorithms

by Carrie Ybay, software engineer at Holberton School

Ongoing project - started 07-15-2020, must end by 07-22-2020 (in 2 days) - you're done with 0% of tasks.

Checker was released at 07-18-2020 12:00 PM

QA review fully automated.

python stack

Requirements

General

- Allowed editors: `vi, vim, emacs`
- All your files will be interpreted/compiled on Ubuntu 14.04 LTS using `python3` (version 3.4.3)
- All your files should end with a new line
- The first line of all your files should be exactly `#!/usr/bin/python3`
- A `README.md` file, at the root of the folder of the project, is mandatory
- Your code should use the `PEP 8` style (version 1.7.x)
- All your files must be executable

Tasks

0. Lockboxes mandatory

Done?

Help

You have `n` number of locked boxes in front of you. Each box is numbered sequentially from `0` to `n - 1` and each box may contain keys to the other boxes.

Write a method that determines if all the boxes can be opened.

- Prototype: `def canUnlockAll(boxes)`
- `boxes` is a list of lists
- A key with the same number as a box opens that box
- You can assume all keys will be positive integers
- The first box `boxes[0]` is unlocked
- Return `True` if all boxes can be opened, else return `False`

```
carrie@ubuntu:~/0x00-lockboxes$ cat main_0.py
#!/usr/bin/python3

canUnlockAll = __import__('0-lockboxes').canUnlockAll

boxes = [[1], [2], [3], [4], []]
print(canUnlockAll(boxes))

boxes = [[1, 4, 6], [2], [0, 4, 1], [5, 6, 2], [3], [4, 1], [6]]
print(canUnlockAll(boxes))

boxes = [[1, 4], [2], [0, 4, 1], [3], [], [4, 1], [5, 6]]
print(canUnlockAll(boxes))

carrie@ubuntu:~/0x00-lockboxes$
```

```
carrie@ubuntu:~/0x00-lockboxes$ ./main_0.py
True
True
False
carrie@ubuntu:~/0x00-lockboxes$
```

Repo:

- GitHub repository: `holbertonschool-interview`
- Directory: `0x00-lockboxes`
- File: `0-lockboxes.py`

Check your code?