Assignments for the 'Introduction to Python' by Stanford University

Code in Place, offered online by Stanford University April-May 2020

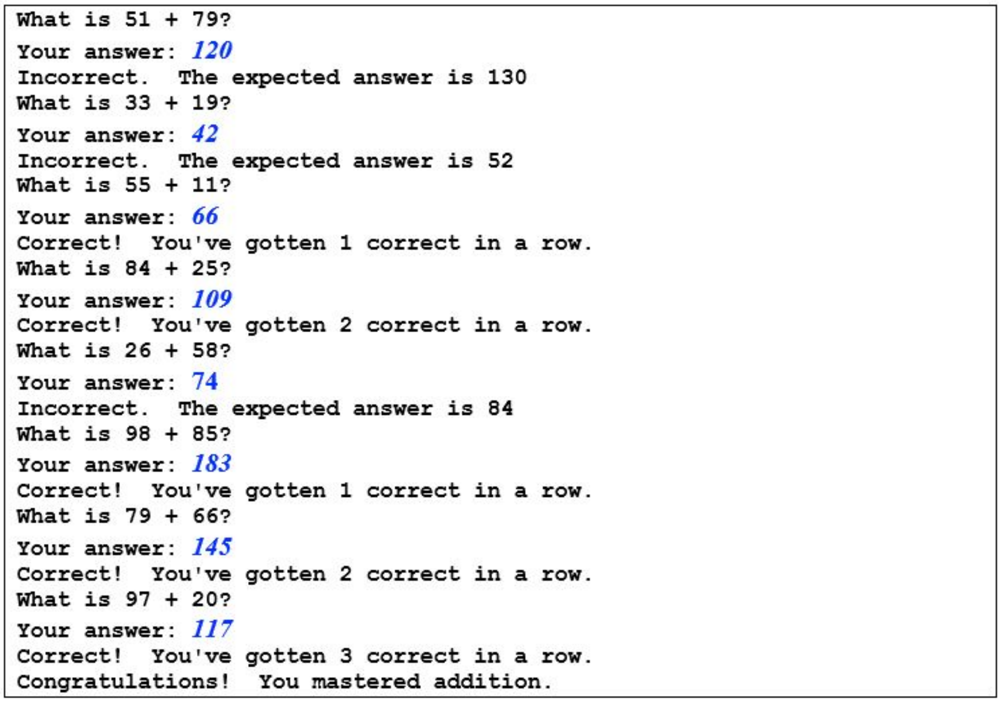
A 5-week introductory online Python programming course based on material from the first half of Stanford’s introductory programming course, CS106A.

Offered by Stanford during COVID-19 pandemic, with 10,000 global students and 900 volunteer teachers participating from around the world.

Below you can see my code for the assignments that I found the most ineteresting.

1) **Khan-Sole Academy**
Khan-sole Academy—a program that helps other people learn. In this problem, you’ll write a program in the file khansole_academy.py that randomly generates simple addition problems for the user, reads in the answer from the user, and then checks to see if they got it right or wrong, until the user appears to have mastered the material.

More specifically, your program should be able to generate simple addition problems that involve adding two 2-digit integers (i.e., the numbers 10 through 99). The user should be asked for an answer to each problem. Your program should determine if the answer was correct or not, and give the user an appropriate message to let them know. Your program should keep giving the user problems until the user has gotten 3 problems correct in a row. (Note: the number of problems the user needs to get correctly in a row to complete the program is just one example of a good place to specify a constant in your program). A sample run of the program is shown below (user input is in italics).



```
What is 51 + 79?
Your answer: 120
Incorrect.  The expected answer is 130
What is 33 + 19?
Your answer: 42
Incorrect.  The expected answer is 52
What is 55 + 11?
Your answer: 66
Correct!  You've gotten 1 correct in a row.
What is 84 + 25?
Your answer: 109
Correct!  You've gotten 2 correct in a row.
What is 26 + 58?
Your answer: 74
Incorrect.  The expected answer is 84
What is 98 + 85?
Your answer: 183
Correct!  You've gotten 1 correct in a row.
What is 79 + 66?
Your answer: 145
Correct!  You've gotten 2 correct in a row.
What is 97 + 20?
Your answer: 117
Correct!  You've gotten 3 correct in a row.
Congratulations!  You mastered addition.
```

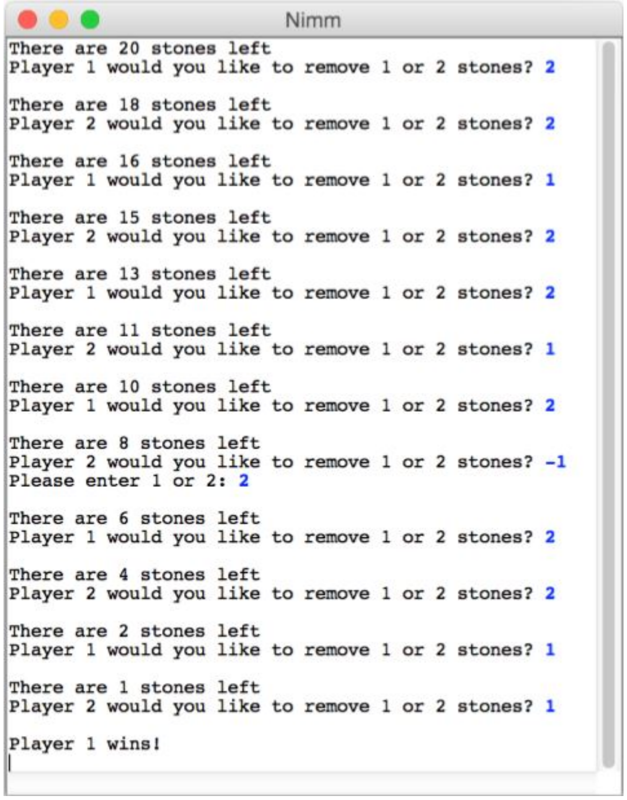
In [ ]:

```
import random

def main():
    corr = 0
    while corr < 3:
        one = random.randint(10,99)
        two = random.randint(10,99)
        correct = one + two
        correct_str = str(correct)
        one_str = str(one)
        two_str = str(two)
        print('What is ' + one_str + ' + ' + two_str + '?')
        input_str = input('Your answer:')
        input_int = int(input_str)

        if input_int != one + two:
            print('Incorrect. The expected answer is ' + correct_str)
        else:
            corr = corr + 1
            corr_str = str(corr)
            print("Correct! You've gotten " + corr_str + " correct in a row.")
    print('Congratulations! You mastered addition.')
```

2) **Nimm**

Nimm is an ancient game of strategy that is named after the old German word for "take." It is also called Tiouk Tiouk in West Africa and Tsynshidzi in China. Players alternate taking stones until there are zero left. The game of Nimm goes as follows:

1. The game starts with a pile of 20 stones between the players 2. The two players alternate turns 3. On a given turn, a player may take either 1 or 2 stone from the center pile 4. The two players continue until the center pile has run out of stones. The last player to take a stone loses.
   Here is a screenshot from an example execution:



```
There are 20 stones left
Player 1 would you like to remove 1 or 2 stones? 2

There are 18 stones left
Player 2 would you like to remove 1 or 2 stones? 2

There are 16 stones left
Player 1 would you like to remove 1 or 2 stones? 1

There are 15 stones left
Player 2 would you like to remove 1 or 2 stones? 2

There are 13 stones left
Player 1 would you like to remove 1 or 2 stones? 2

There are 11 stones left
Player 2 would you like to remove 1 or 2 stones? 1

There are 10 stones left
Player 1 would you like to remove 1 or 2 stones? 2

There are 8 stones left
Player 2 would you like to remove 1 or 2 stones? -1
Please enter 1 or 2: 2

There are 6 stones left
Player 1 would you like to remove 1 or 2 stones? 2

There are 4 stones left
Player 2 would you like to remove 1 or 2 stones? 2

There are 2 stones left
Player 1 would you like to remove 1 or 2 stones? 1

There are 1 stones left
Player 2 would you like to remove 1 or 2 stones? 1

Player 1 wins!
```

In [ ]:

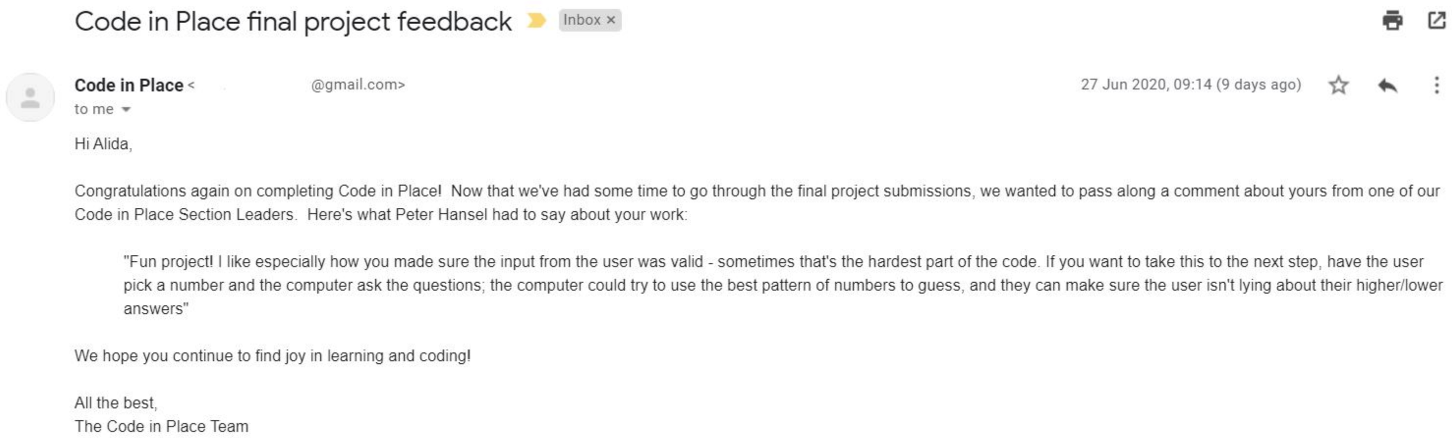
```
def main():
    stones=20
    player= 1
    while stones > 0:
       stones_str = str(stones)
       print("There are " + stones_str + " stones left")
       if player == 1:
           input_int = int(input("Player 1 would you like to remove 1 or 2 stones? "))
           while input_int > 2:
            input_int = int(input("Please enter 1 or 2: "))
           stones = stones - input_int
           player = player + 1
           player = player + 1
       else:
           input_int = int(input("Player 2 would you like to remove 1 or 2 stones? "))
           while input_int > 2:
               input_int = int(input("Please enter 1 or 2: "))
           stones = stones - input_int
           player = player - 1
    if player == 2:
        print("Player 2 wins!")
    else:
        print("Player 1 wins!")
```

3) **Guess the number**

This is a game where a player is given 5 tries to guess a random number between 0 and 50.

In [ ]:

```
import random
random_num = random.randint(0,50)
#print(random_num)
def main():
    tries = 5
    print("You need to guess the number from 0 to 50. You've got 5 tries. Good luck!")


    while tries > 0:
        inp_num = int(input('Your guess: '))
        if inp_num < random_num:
            tries = tries - 1
            print('Wrong answer. The number is greater than ' + str(inp_num) + ". You've
got "+ str(tries) + ' guesses left.')
        elif inp_num > random_num:
            while inp_num > 50:
                inp_num = int(input("Please enter a number between 0 and 50: "))
            tries = tries - 1
            print('Wrong answer. The number is smaller than ' + str(inp_num) + ". You've
got "+ str(tries) + ' guesses left. ')
        else:
            print('Spot on!')
            exit(0)
    print("Sorry, you lost")
```



Code in Place final project feedback Inbox ×

Code in Place < @gmail.com>
to me

27 Jun 2020, 09:14 (9 days ago)

Hi Alida,

Congratulations again on completing Code in Place! Now that we've had some time to go through the final project submissions, we wanted to pass along a comment about yours from one of our Code in Place Section Leaders. Here's what Peter Hansel had to say about your work:

> "Fun project! I like especially how you made sure the input from the user was valid - sometimes that's the hardest part of the code. If you want to take this to the next step, have the user pick a number and the computer ask the questions; the computer could try to use the best pattern of numbers to guess, and they can make sure the user isn't lying about their higher/lower answers"

We hope you continue to find joy in learning and coding!

All the best,
The Code in Place Team