CSE 231 Spring 2013

Programming Project 02

Assignment Overview

This project will give you practice using operations on integers, conditions, and selection (*if*). This assignment is worth 20 points (2% of the course grade). It must be **completed and turned in before 11:59 on Monday, January 21, 2013.**

One strategy for finding a solution to a number puzzle is to describe the solution by a set of algebraic constraints and then solve the constraints. Consider, for instance, the following puzzle:

The Jets and the Sharks are the names of two rival dance teams. The Jets say to the Sharks: If one of you joins our team, then our team will be double the size of yours. The Sharks reply: If one of you joins our team, then the sizes of our teams will be equal. What are the sizes of the two teams?

If the sizes of the Jets and Sharks are denoted by j and s, respectively, then any solution to this puzzle satisfies the constraints: $(j + 1) = 2(s - 1)$ and $s + 1 = j - 1$. Solving these constraints yields a solution to the puzzle, namely, $j = 7$ and $s = 5$.

While solving the constraints in this example is easy, solving constraints describing some number puzzles can be too hard. In such cases, another strategy is to guess possible solutions and then check if the guess works. Consider, for instance, the following puzzle:

What digits can replace the letters A, B and C to make a 3-digit number ABC for which the following equation is true: $ABC = A{\cdot}B{\cdot}C{\cdot}(A+B+C)$

This puzzle is not easily solved directly. But any guess for the 3-digit number *ABC* can be easily checked to see if it satisfies the equation. For example, suppose you guess that *ABC* is 123. This guess does not work because it requires $A = 1$, $B = 2$, and $C = 3$, and $123 \neq 1{\cdot}2{\cdot}3{\cdot}\ (1+2+3)$. In contrast, the guess that *ABC* is 135 works since $135 = 1{\cdot}3{\cdot}5{\cdot}(1+3+5)$.

You will write a program that can be used to check guesses for the following puzzle:

For what six-digit number SLAYER is the following equation true, where each letter stands for the digit in the position shown: SLAYER + SLAYER + SLAYER = LAYERS

Project Description / Specification

Your program must:

1. Output a brief descriptive message (see example below).
2. Prompt for a guess for SLAYER.
3. Check if the guess solves the puzzle and print the result of the check along with a brief justification.

To clarify these specifications, the end of this write-up contains output produced by a program that meets them.

Programming requirements:

1. Although this problem can be solved using strings and string operations, to reinforce your understanding of numeric expressions and the representation of integers, you are to solve it

using integers and integer operations instead. (See the **Notes and Hints** below.) We will study strings and string operations later in the semester.

2. Include the following types of blocks where appropriate: *if* and *else*.
3. Include some chained relational expressions where appropriate.

Deliverables

`proj02.py` – your source code solution (*remember to include your section, the date, project number and comments in this file;* ***do not*** *include your PID or name*).

1. Be sure to use "`proj02.py`" for the file name (or handin might not accept it!)
2. Save a copy of your file in your CSE account disk space (H drive on CSE computers). ***This is the only way we can check that you completed the project on time in case you have a problem with handin.***
3. Electronically submit a copy of the file using handin.

Notes and Hints:

1. You do not need to check that the user enters an integer—i.e., your program does not need to work correctly if the user enters anything but an integer at the prompt.

2. The quotient (//) and remainder (%) operations are useful for extracting the digits from an integer. For example:

```
>>> num = 2013
>>> num%10    # the digit in the ones-place
3
>>> num//10   # the number of tens in num
201
>>> (num//10)%10  # the digit in the tens-place
1
>>> num//100  # the number of hundreds in num
20
>>> (num//100)%10  # the digit the hundreds-place
0
>>> num//1000  # the number of thousands in num
2
>>> (num//1000)%10  # the digit in the thousands-place
2
>>>
```

3. Decimal notation tells you how to calculate an integer if you know its digits and their positions. For example: given digits A, B, and C, the integer represented by ABC is 100*A + 10*B + C.

4. Save your work early and often. You can save by submitting it to handin or by saving it to your CSE file space. This is the only way to ensure that you do not lose hours of work if your Python session crashes and that you can receive some credit for work done if you experience technical difficulties (e.g., a network or power failure) close to the handin deadline.

Questions for your to consider (not hand in):

1. Guessing a correct solution to this puzzle is not easy because there are so many possible guesses. Now that you know about iteration, how might you use it to write a program that finds all solution(s) to this puzzle without a user needing to provide guesses?
2. What happens if a user does not enter an integer when prompted for a guess? How could you fix this problem?

Examples:

To illustrate, we show results of executing a program that meets the project specifications for a number of different test inputs. In these examples, lines that start with >>> were produced by the Python Shell (by the F5 command, which runs the program). Everything else was produced by print statements in the program except for the characters shown in red. The test inputs that we used (the responses we typed at the prompts) are shown in red.

The first two examples show results of guessing incorrect solutions; the last shows results of guessing a correct one.

```
>>> =========================== RESTART ============================
>>> 

Guess a six-digit number SLAYER so that following equation is true,
where each letter stands for the digit in the position shown:

SLAYER + SLAYER + SLAYER = LAYERS


Enter your guess for SLAYER: 9015
Your guess is incorrect:
SLAYER must be a 6-digit number.
Thanks for playing.

>>> =========================== RESTART ============================
>>> 

Guess a six-digit number SLAYER so that following equation is true,
where each letter stands for the digit in the position shown:

SLAYER + SLAYER + SLAYER = LAYERS


Enter your guess for SLAYER: 666666
Your guess is incorrect:
SLAYER + SLAYER + SLAYER = 1999998
LAYERS = 666666
Thanks for playing.

>>> =========================== RESTART ============================
>>> 

Guess a six-digit number SLAYER so that following equation is true,
where each letter stands for the digit in the position shown:

SLAYER + SLAYER + SLAYER = LAYERS


Enter your guess for SLAYER: 142857
Your guess is correct:
```

```
SLAYER + SLAYER + SLAYER = 428571
LAYERS = 428571
Thanks for playing.
```