

Programming Project #4

Assignment Overview

This project focuses on string manipulation, as well as more experience with control. It is due Monday, October 12th before midnight. That is two weeks, longer than normal, because of the exam on October 8th. If you can, you should consider getting it done before the normal deadline. Up to you.

The Problem

The **196-algorithm** is a procedure for creating a palindromic integer: an integer that has the same value when examined forwards or backwards. Examples of palindromic integers are: 88, 121, 2332, 12321, etc. The 196-algorithm is as follows.

1. If the integer is a palindrome, then print that integer
2. Otherwise, take the integer and its reversal and add them together.
3. With the sum, repeat the process starting at step 1.

Here are some examples:

56: $56+65=121$! palindrome

57: $57+75=132$; $132+231=363$! palindrome

87: $87+78=165$; $165+561=726$; $726+627=1353$; $1353+3531=4884$! palindrome

89: 24 steps to the palindromic number 8813200023188

It is called the 196-algorithm because the integer 196 is the first number that, it appears, does not converge to a palindromic number. Such a number is called a **Lychrel** number (see http://en.wikipedia.org/wiki/Lychrel_number). Though it hasn't been mathematically **proven** that 196 doesn't converge, it has been shown to **not** converge out to a number containing 300 million digits!

Program Specifications

Your project will do the following:

1. Prompt for two integers. These two integers constitute the range of integers (inclusive) of the integers that will be checked.
2. After the program runs, you will report the following statistics for the numbers examined in the given range:
 - a. The number of "natural" palindromes (numbers in the range that are already palindromes, such as 11, 121, 3553)
 - b. The number of non-Lychrel numbers (numbers which eventually yield a palindrome using the 196 algorithm)
 - c. The number of Lychrel numbers encountered. Assume a maximum of 60 iterations to indicate a Lychrel number
 - d. Because Lychrel numbers are rare, report each Lychrel number as it occurs.

Deliverables

proj04.py -- your source code solution (remember to include your section, the date, project number and comments).

1. Please be sure to use the specified file name, i.e. "proj04.py"

2. Save a copy of your file in your CS account disk space (H drive on CS computers).
3. Electronically submit a copy of the file.

Assignment Notes:

One of the main issues is converting values back and forth between integers and strings (using the `int()` and `str()` functions). Determinations of palindrome and number reversal should be done as strings, while clearly addition must be done with integers.

You are free to use any of the code we showed in class concerning palindromes. Remember that there are no issues about capitalization or non-characters in this example since we are restricted to dealing with sequences of integers.

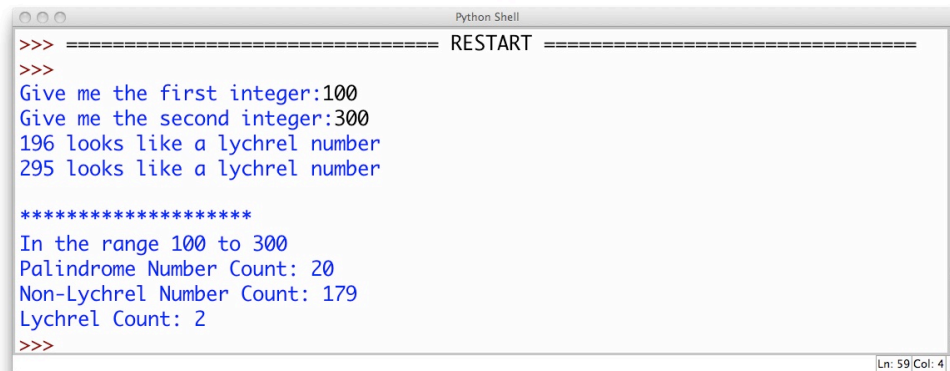
Remember that python has no restriction on the size of an integer.

To clarify the problem specifications, we provide at the end of this document a snapshot of interaction with the already written program.

Getting Started

1. Using IDLE create a new program.
2. If you are in a CSE lab, select the H: drive as the location to store your file
3. Save the name of the project: proj04.py
4. Create the preface print information and prompt for the user integers
5. Run the program and fix any errors
6. Use the web site to hand in the program (incomplete as this point but you should continually hand things in)
7. Now, take the user number and check if it is a palindrome
8. Again, if you solve that save the program, as well as hand it in.
9. Now implement the reverse and add process. Combine this with the palindrome check.
10. Now see if you can do this iteratively for a range of numbers
11. Finally, check the counts at the end and print out the results.
12. Now you enter a cycle of edit-run to incrementally develop your program.
13. Hand in your final version.

Sample Interaction

A screenshot of a Python Shell window titled "Python Shell". The window shows the following text: a prompt ">>>" followed by a separator line "===== RESTART =====", another prompt ">>>", and then the program's output: "Give me the first integer:100", "Give me the second integer:300", "196 looks like a lychrel number", and "295 looks like a lychrel number". After a separator line "*****", it shows "In the range 100 to 300", "Palindrome Number Count: 20", "Non-Lychrel Number Count: 179", and "Lychrel Count: 2". The prompt ">>>" appears at the bottom. The status bar at the bottom right indicates "Ln: 59 Col: 4".

```
>>> ===== RESTART =====
>>>
Give me the first integer:100
Give me the second integer:300
196 looks like a lychrel number
295 looks like a lychrel number

*****
In the range 100 to 300
Palindrome Number Count: 20
Non-Lychrel Number Count: 179
Lychrel Count: 2
>>>
```