Họ và tên: NGUYỄN ĐÌNH TIẾN

MSSV: 20168533

Lớp: VUWIT15

Project 1

Program Language: Python

Tuần: 11

Project Euler

Problem 53:

1.Đề bài: There are exactly ten ways of selecting three from five, 12345:

123, 124, 125, 134, 135, 145, 234, 235, 245, and 345

In combinatorics, we use the notation, 5C3 = 10.

In general,

|  |  |  |
| --- | --- | --- |
| *n*C*r* = | *n*! *r*!(*n−r*)! | ,where *r* ≤ *n*, *n*! = *n*×(*n*−1)×...×3×2×1, and 0! = 1. |

It is not until *n* = 23, that a value exceeds one-million: 23C10 = 1144066.

How many, not necessarily distinct, values of  *n*C*r*, for 1 ≤ *n* ≤ 100, are greater than one-million?

2.Thuật toán:

- Viết hàm tính tổ hợp

- xét giá trị n trong đoạn 23,100 (vì đến 23 mới có giá trị thỏa mãn)

- Tính tổ hợp cnr với từng giá trị n trong khoảng đã cho và r là giá trị trong đoạn 1,n, nếu tổ hợp lớn hơn 1 triệu thì đếm count thêm 1

- Gía trị count sau khi duyệt hết n là kết quả cần tìm

3.Đáp án: 4075

4.Mã nguồn:

# Combinatoric selections

from math import factorial as f

count = 0

def ncr(n, r):

return f(n)/(f(r)\*f(n-r))

for n in range(23, 101):

for r in range(1, n):

if ncr(n, r) > 1000000:

count += 1

print(count)

Problem 55:

1.Đề bài:

If we take 47, reverse and add, 47 + 74 = 121, which is palindromic.

Not all numbers produce palindromes so quickly. For example,

349 + 943 = 1292,  
1292 + 2921 = 4213  
4213 + 3124 = 7337

That is, 349 took three iterations to arrive at a palindrome.

Although no one has proved it yet, it is thought that some numbers, like 196, never produce a palindrome. A number that never forms a palindrome through the reverse and add process is called a Lychrel number. Due to the theoretical nature of these numbers, and for the purpose of this problem, we shall assume that a number is Lychrel until proven otherwise. In addition you are given that for every number below ten-thousand, it will either (i) become a palindrome in less than fifty iterations, or, (ii) no one, with all the computing power that exists, has managed so far to map it to a palindrome. In fact, 10677 is the first number to be shown to require over fifty iterations before producing a palindrome: 4668731596684224866951378664 (53 iterations, 28-digits).

Surprisingly, there are palindromic numbers that are themselves Lychrel numbers; the first example is 4994.

How many Lychrel numbers are there below ten-thousand?

NOTE: Wording was modified slightly on 24 April 2007 to emphasise the theoretical nature of Lychrel numbers.

2. Thuật toán:

- Viết hàm kiểm tra số lychrel (Đảo ngược 1 số cộng lại và kiểm tra kết quả có phải số đối xứng hay không, được lặp lại tối đa 50 lần nếu không thể được số đối xứng thì chính là số lycherel)

- Duyệt các số nhỏ hơn 10001, số nào là số lychrel thì count cộng thêm 1

- Sau khi duyệt xong thì count chính là số các số lychrel không lớn hơn 10000 là kết quả cần tìm

3.Đáp án: 249

4.Mã nguồn:

# Lychrel numbers

def is\_lychrel(n):

for i in range(50):

number = n + int(str(n)[::-1])

if str(number) == str(number)[::-1]:

return False

n = number

return True

count = 0

for i in range(10001):

if is\_lychrel(i):

count += 1

print(count)

Problem 56:

1.Đề bài:

A googol (10100) is a massive number: one followed by one-hundred zeros; 100100 is almost unimaginably large: one followed by two-hundred zeros. Despite their size, the sum of the digits in each number is only 1.

Considering natural numbers of the form, *ab*, where *a, b* < 100, what is the maximum digital sum?

2.Thuật toán:

- Viết hàm tính tổng các chữ số của một số

- Xét a, b < 100, tính *ab* và tính tổng các chứ số của giá trị đó, nếu tổng lớn hơn giá trị largest lớn nhất hiện tại thì gán largest bằng giá trị đó

- Gía trị largest sau khi xét xong chính là kết quả cần tìm

3.Đáp án: 972

4.Mã nguồn:

# Powerful digit sum

def sum\_of\_digits(n):

sod = 0

while n != 0:

sod += n % 10

n //= 10

return sod

largest = 0

for a in range(0, 100):

for b in range(0, 100):

sod = sum\_of\_digits(a\*\*b)

if sod > largest:

largest = sod

print(largest)