Họ và tên: NGUYỄN ĐÌNH TIẾN

MSSV: 20168533

Lớp: VUWIT15

Project 1

Program Language: Python

Tuần: 7

Project Euler

Problem 28:

1.Đề bài:

Starting with the number 1 and moving to the right in a clockwise direction a 5 by 5 spiral is formed as follows:

**21** 22 23 24 **25**  
20  **7**  8  **9** 10  
19  6  **1**  2 11  
18  **5**  4  **3** 12  
**17** 16 15 14 **13**

It can be verified that the sum of the numbers on the diagonals is 101.

What is the sum of the numbers on the diagonals in a 1001 by 1001 spiral formed in the same way?

2.Thuật toán:

- Dùng vòng for để tính tổng tất cả các số nằm trên đường chéo vì ta thấy các số sau bằng số trước đó cộng thêm độ dài cạnh trừ đi 1

3.Đáp án: 669171001

4.Mã nguồn:

# What is the sum of the numbers on the diagonals in a 1001 by 1001 spiral formed in the same way?

sum = 25 # side = 3

last\_number = 9

for i in range(3, 1000, 2):

for j in range(4):

last\_number += i + 1

sum += last\_number

print(sum)

Problem 29:

1.Đề bài:

Consider all integer combinations of *ab* for 2 ≤ *a* ≤ 5 and 2 ≤ *b* ≤ 5:

22=4, 23=8, 24=16, 25=32  
32=9, 33=27, 34=81, 35=243  
42=16, 43=64, 44=256, 45=1024  
52=25, 53=125, 54=625, 55=3125

If they are then placed in numerical order, with any repeats removed, we get the following sequence of 15 distinct terms:

4, 8, 9, 16, 25, 27, 32, 64, 81, 125, 243, 256, 625, 1024, 3125

How many distinct terms are in the sequence generated by *ab* for 2 ≤ *a* ≤ 100 and 2 ≤ *b* ≤ 100?

2. Thuật toán:

- Dùng vòng for duyệt các giá trị a và b trong khoảng đã cho và kiểm tra nếu giá trị đó chưa được tính thì ta thêm vào list sau đó tìm độ dài list ta được giá trị cần tìm

3.Đáp án: 9183

4.Mã nguồn:

# Có bao nhiêu số nằm trong chuỗi tạo thành từ a^b với 2 <= a,b <= 100

list\_terms = []

for a in range(2, 101):

for b in range(2,101):

if a \*\* b not in list\_terms:

list\_terms.append(a\*\*b)

print(len(list\_terms))

Problem 35:

1.Đề bài:

The number, 197, is called a circular prime because all rotations of the digits: 197, 971, and 719, are themselves prime.

There are thirteen such primes below 100: 2, 3, 5, 7, 11, 13, 17, 31, 37, 71, 73, 79, and 97.

How many circular primes are there below one million?

2.Thuật toán:

- Comment trên mã nguồn bên dưới

3.Đáp án: 55

4.Mã nguồn:

# # How many circular primes are there below one million?

from itertools import permutations

from math import ceil

list\_primes = []

# Tìm tất cả số nguyên tố dưới 1 triệu

n = 1000000

primes = [True] \* n

primes[0] = False

primes[1] = False

roundUp = lambda n, prime: int(ceil(n / prime))

for currentPrime in range(2, n):

if not primes[currentPrime]:

continue

list\_primes.append(currentPrime)

for i in range(2, roundUp(n, currentPrime)):

primes[i \* currentPrime] = False

# Kiểm tra tất cả các số nguyên tố có đảo không là số nguyên tố và thêm vào 1 list

list\_remove = []

for number\_check in list\_primes:

print(len(list\_remove))

if ("2" in str(number\_check)) or ("4" in str(number\_check)) or ("6" in str(number\_check)) or ("8" in str(number\_check)) or ("0" in str(number\_check)) or ("5" in str(number\_check)):

list\_remove.append(number\_check)

continue

number = str(number\_check)

for i in range(0, len(number)):

rotatedNumber = number[i:len(number)] + number[0:i]

if int(rotatedNumber) not in list\_primes:

list\_remove.append(number\_check)

# Remove tất cả số nguyên tố đảo đi thì không là số nguyên tố ra khỏi list tất cả số nguyên tố

for remove\_number in list\_remove:

if remove\_number in list\_primes:

list\_primes.remove(remove\_number)

# Trong quá trình lọc thì lọc cả số 2 và 5 thỏa mãn nên ta thêm lại

list\_primes.append(2)

list\_primes.append(5)

print(len(list\_primes))