Programming Assignment_9

1. Write a Python program to check if the given number is a Disarium Number?

In [6]:
```
def is_disarium(n):
    # Convert the number to a string
    num_str = str(n)
    # Initialize a variable to store the sum
    sum = 0
    # Iterate over the digits of the number
    for i, digit in enumerate(num_str, start=1):
        # Add the digit powered to its position to the sum
        sum += int(digit) ** i
    # Return True if the sum is equal to the original number, False otherwise
    return sum == n
print(is_disarium(135)) # True
print(is_disarium(175)) # True
print(is_disarium(89)) # False
print(is_disarium(547)) # False
```

```
True
True
True
False
```

2. Write a Python program to print all disarium numbers between 1 to 100?

In [7]:
```
def print_disarium_numbers():
    # Iterate over the numbers from 1 to 100
    for n in range(1, 101):
        # Check if the number is a Disarium number
        if is_disarium(n):
            # Print the number if it is a Disarium number
            print(n)
def is_disarium(n):
    # Convert the number to a string
    num_str = str(n)
    # Initialize a variable to store the sum
    sum = 0
    # Iterate over the digits of the number
    for i, digit in enumerate(num_str, start=1):
        # Add the digit powered to its position to the sum
        sum += int(digit) ** i
    # Return True if the sum is equal to the original number, False otherwise
    return sum == n
# Print all Disarium numbers between 1 and 100
print_disarium_numbers()
```

```
1
2
3
4
5
6
7
8
9
89
```

3. Write a Python program to check if the given number is Happy Number?

In [8]:
```python
def is_happy(n):
    # Initialize a set to store the numbers that have been seen
    seen = set()
    # Repeat the process until the number is 1 or it has been seen before
    while n != 1 and n not in seen:
        # Add the number to the set of seen numbers
        seen.add(n)
        # Replace the number with the sum of the squares of its digits
        n = sum(int(digit) ** 2 for digit in str(n))
    # Return True if the number is 1, False otherwise
    return n == 1
print(is_happy(7)) # True
print(is_happy(13)) # True
print(is_happy(4)) # False
print(is_happy(9)) # False
```

```
True
True
False
False
```

4. Write a Python program to print all happy numbers between 1 and 100?

In [9]:
```python
def print_happy_numbers():
    # Iterate over the numbers from 1 to 100
    for n in range(1, 101):
        # Check if the number is a happy number
        if is_happy(n):
        # Print the number if it is a happy number
            print(n)
def is_happy(n):
    # Initialize a set to store the numbers that have been seen
    seen = set()
    # Repeat the process until the number is 1 or it has been seen before
    while n != 1 and n not in seen:
        # Add the number to the set of seen numbers
        seen.add(n)
        # Replace the number with the sum of the squares of its digits
        n = sum(int(digit) ** 2 for digit in str(n))
    # Return True if the number is 1, False otherwise
    return n == 1
```

```
# Print all happy numbers between 1 and 100
print_happy_numbers()
```

```
1
7
10
13
19
23
28
31
32
44
49
68
70
79
82
86
91
94
97
100
```

5. Write a Python program to determine whether the given number is a Harshad Number?

In [11]:

```
def calculate_sum_of_digits(N):
    sumOfDigits = 0
    while N > 0:
        digit = N % 10
        sumOfDigits = sumOfDigits + digit
        N = N // 10
    return sumOfDigits


input_number = 4320
output = calculate_sum_of_digits(input_number)
print("Sum of digits of {} is {}.".format(input_number, output))
def calculate_sum_of_digits(N):
    sumOfDigits = 0
    while N > 0:
        digit = N % 10
        sumOfDigits = sumOfDigits + digit
        N = N // 10
    return sumOfDigits


def check_for_harshad_number(N):
    sumOfDigits = calculate_sum_of_digits(N)
    if N % sumOfDigits == 0:
        return True
    else:
        return False


input_number = 4320
output = check_for_harshad_number(input_number)
print("{} is a Harshad Number:{}".format(input_number, output))
```

```
input_number = 4321
output = check_for_harshad_number(input_number)
print("{} is a Harshad Number:{}".format(input_number, output))
```

```
Sum of digits of 4320 is 9.
4320 is a Harshad Number:True
4321 is a Harshad Number:False
```

In [ ]:

6. Write a Python program to print all pronic numbers between 1 and 100?

In [12]:

```
def is_pronic(n):
    # Iterate over the integers from 1 to the square root of the number
    for i in range(1, int(n ** 0.5) + 1):
        # Check if the number is the product of two consecutive integers
        if n == i * (i + 1):
            return True
    return False
# Iterate over the numbers from 1 to 100
for n in range(1, 101):
    # Check if the number is a pronic number
    if is_pronic(n):
        # Print the number if it is a pronic number
        print(n)
```

```
2
6
12
20
30
42
56
72
90
```

In [ ]: