**ADITYA Amin**

**Assign : 09**

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

def is\_disarium(number):

"""

Checks if a given number is a Disarium number and returns True or False.

"""

# Convert the number to string to determine its length

num\_str = str(number)

# Calculate the sum of each digit raised to its position

sum = 0

for i, digit\_str in enumerate(num\_str, start=1):

digit = int(digit\_str)

sum += digit \*\* i

# Check if the calculated sum is equal to the original number

if sum == number:

return True

else:

return False

# Example number to check

num = 135

# Call the function to check if the number is a Disarium number

if is\_disarium(num):

print(f"{num} is a Disarium number.")

else:

print(f"{num} is not a Disarium number.")

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

def is\_disarium(number):

"""

Checks if a given number is a Disarium number and returns True or False.

"""

# Convert the number to string to determine its length

num\_str = str(number)

# Calculate the sum of each digit raised to its position

sum = 0

for i, digit\_str in enumerate(num\_str, start=1):

digit = int(digit\_str)

sum += digit \*\* i

# Check if the calculated sum is equal to the original number

if sum == number:

return True

else:

return False

# Loop through numbers from 1 to 100

for num in range(1, 101):

# Call the function to check if the number is a Disarium number

if is\_disarium(num):

print(f"{num} is a Disarium number.")

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

def is\_happy\_number(number):

"""

Checks if a given number is a Happy number and returns True or False.

"""

# Create a set to keep track of visited numbers

visited = set()

# Loop until the number becomes 1 (a Happy number) or repeats (not a Happy number)

while number not in visited:

# Add the current number to the set of visited numbers

visited.add(number)

# Calculate the sum of squares of its digits

sum\_of\_squares = 0

num\_str = str(number)

for digit\_str in num\_str:

digit = int(digit\_str)

sum\_of\_squares += digit \*\* 2

# Update the number with the sum of squares

number = sum\_of\_squares

# If the number becomes 1, it's a Happy number

if number == 1:

return True

else:

return False

# Example number to check

num = 19

# Call the function to check if the number is a Happy number

if is\_happy\_number(num):

print(f"{num} is a Happy number.")

else:

print(f"{num} is not a Happy number.")

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

def is\_happy\_number(number):

"""

Checks if a given number is a Happy number and returns True or False.

"""

# Create a set to keep track of visited numbers

visited = set()

# Loop until the number becomes 1 (a Happy number) or repeats (not a Happy number)

while number not in visited:

# Add the current number to the set of visited numbers

visited.add(number)

# Calculate the sum of squares of its digits

sum\_of\_squares = 0

num\_str = str(number)

for digit\_str in num\_str:

digit = int(digit\_str)

sum\_of\_squares += digit \*\* 2

# Update the number with the sum of squares

number = sum\_of\_squares

# If the number becomes 1, it's a Happy number

if number == 1:

return True

else:

return False

# Loop through numbers from 1 to 100

for num in range(1, 101):

# Call the function to check if the number is a Happy number

if is\_happy\_number(num):

print(f"{num} is a Happy number.")

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

def is\_harshad\_number(number):

"""

Checks if a given number is a Harshad number and returns True or False.

"""

# Convert the number to string to calculate the sum of its digits

num\_str = str(number)

# Calculate the sum of digits

sum\_of\_digits = sum(int(digit) for digit in num\_str)

# Check if the number is divisible by the sum of its digits

if number % sum\_of\_digits == 0:

return True

else:

return False

# Example number to check

num = 18

# Call the function to check if the number is a Harshad number

if is\_harshad\_number(num):

print(f"{num} is a Harshad number.")

else:

print(f"{num} is not a Harshad number.")

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

def is\_pronic\_number(number):

"""

Checks if a given number is a pronic number and returns True or False.

"""

# Loop through numbers from 1 to n to find the pronic number

for i in range(1, number+1):

# Calculate the product of consecutive numbers

product = i \* (i + 1)

# Check if the product matches the given number

if product == number:

return True

# If no match found, it's not a pronic number

return False

# Loop through numbers from 1 to 100

for num in range(1, 101):

# Call the function to check if the number is a pronic number

if is\_pronic\_number(num):

print(f"{num} is a pronic number.")