Programming Assignment-9

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

Ans.

```
def is_disarium(num):
    num_str = str(num)
    return num == sum(int(digit) ** (i + 1) for i, digit in enumerate(num_str))

# Example
num = 89
print(f"{num} is a Disarium number? {is_disarium(num)}")
```

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

Ans.

```
def is_disarium(num):
    num_str = str(num)
    return num == sum(int(digit) ** (i + 1) for i, digit in enumerate(num_str))

def disarium_numbers_in_range(start, end):
    return [num for num in range(start, end + 1) if is_disarium(num)]

# Example
print("Disarium numbers between 1 and 100:", disarium_numbers_in_range(1, 100))
```

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

Ans.

```
def is_happy_number(num):
    seen = set()
    while num != 1 and num not in seen:
        seen.add(num)
```

```
        num = sum(int(digit) ** 2 for digit in str(num))
    return num == 1

# Example
num = 19
print(f"{num} is a Happy number? {is_happy_number(num)}")
```

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

Ans.

```
def is_happy_number(num):
    seen = set()
    while num != 1 and num not in seen:
        seen.add(num)
        num = sum(int(digit) ** 2 for digit in str(num))
    return num == 1

def happy_numbers_in_range(start, end):
    return [num for num in range(start, end + 1) if is_happy_number(num)]

# Example
print("Happy numbers between 1 and 100:", happy_numbers_in_range(1, 100))
```

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

Ans.

```
def is_harshad_number(num):
    return num % sum(int(digit) for digit in str(num)) == 0

# Example
num = 18
```

```
print(f"{num} is a Harshad number? {is_harshad_number(num)}")
```

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

Ans.

```
def pronic_numbers_in_range(start, end):
    pronic_numbers = []
    for n in range(1, int(end ** 0.5) + 1):
        pronic_num = n * (n + 1)
        if pronic_num >= start and pronic_num <= end:
            pronic_numbers.append(pronic_num)
    return pronic_numbers

# Example
print("Pronic numbers between 1 and 100:", pronic_numbers_in_range(1, 100))
```