

```
In [1]: import math
```

```
class LehmerRandomNumberGenerator:
    """
    This class is responsible for generating uniformly distributed random numbers
    on the interval (0, 1)
    """
    def __init__(self, seed: int, modulus: int, multiplier: int):
        """
        The constructor of the class
        :param seed: the seed to start with
        :param modulus: the modulus value
        :param multiplier: the multiplier
        """
        self.last_value = seed
        self.modulus = modulus
        self.multiplier = multiplier

    def next(self) -> float:
        """
        Returns the next random number in the sequence
        :return:
        """
        self.last_value = (self.multiplier * self.last_value) % self.modulus
        return self.last_value / self.modulus

def find_period_multiplier(modulus: int) -> int:
    """
    Finds a full-period multiplier for the given modulus
    :param modulus: the modulus to check
    :return: a full-period multiplier
    """
    a = 1

    while True:
        p = 1
        x = a

        while x != 1:
            p += 1
            x = (a * x) % modulus

        if p == modulus - 1:
            print("{} is a full period multiplier\n".format(a))
            return a

        a += 1

def find_all_multipliers(modulus: int, multiplier: int = None) -> list:
    """
    Finds all full-period multiplier for the given modulus and a multiplier
    :param modulus: the modulus to check
    :param multiplier: the multiplier to start with
    :return: a list of all full-period multipliers
    """
    if multiplier is None:
        multiplier = find_period_multiplier(modulus)

    i = 1
    x = multiplier
    multipliers = []

    while x != 1:
        if math.gcd(i, modulus - 1) == 1:
            multipliers.append(x)
            i += 1
            x = (multiplier * x) % modulus

    multipliers.sort()
    return multipliers

if __name__ == '__main__':
    mod = 251
    multiplier_list = find_all_multipliers(mod)
    print("Found {} Period Multipliers of {}".format(len(multiplier_list), mod))
    for m in multiplier_list:
        print(m)
```

6 is a full period multiplier

Found 100 Period Multipliers of 251:

- 6
- 11
- 14
- 18
- 19
- 24
- 26
- 29
- 30
- 33
- 34
- 37
- 42
- 43
- 44
- 46
- 53
- 54
- 55
- 56
- 57
- 59
- 61
- 62
- 70
- 71
- 72
- 76
- 77
- 78
- 82
- 87
- 90
- 95
- 96
- 97
- 98
- 99
- 104
- 107
- 109
- 111
- 116
- 120
- 127
- 129
- 130
- 132
- 133
- 134
- 136
- 137
- 139
- 141
- 143
- 145
- 146
- 148
- 150
- 158
- 159
- 162
- 163
- 165
- 166
- 167
- 168
- 170
- 172
- 176
- 177
- 178
- 183
- 184
- 185
- 186
- 191
- 193

199  
202  
203  
206  
210  
212  
213  
215  
216  
220  
223  
224  
228  
229  
230  
234  
236  
238  
239  
242  
244  
248

In [ ]:

In [ ]:

In [ ]: