

メモ : $L(x) = n$ なる集合

2019.01.21 山下 倫範, 宮田大輔, 藤田菜摘

2019.02.18 update

L を Euler's torsion 関数の導来対数関数とし, $M(n) = \{x \in \mathbb{N} \mid L(x) = n\}$, $m(n) = |M(n)|$, $m(0) = 1$ とする。

さらに,

$$M_e(n, t) = \{x : \text{even} \mid x = 2^t x_0, x_0 : \text{odd} \mid L(x) = n\}$$

$$M_e(n) = \{x : \text{even} \mid L(x) = n\}$$

$$M_e(n) = \bigcup_{k=1}^n M_e(n, k),$$

$$M_p(n) = \{x : \text{prime} \mid L(x) = n\}, \quad c_n = |M_p(n)| \text{ としておく。}$$

以下簡単に次のように計算できる。

$$M_p(1) = \{2, 3\}$$

$$M_p(2) = \{5, 7\}$$

$$M_p(3) = \{11, 13, 19\}$$

$$M_p(4) = \{17, 23, 29, 31, 37, 43\}$$

$$M_p(5) = \{41, 47, 53, 59, 61, 67, 71, 73, 79, 109, 127, 163\}$$

$$|M_e(n)| = \sum_{k=1}^n |M_e(n, k)|$$

$$M_p(6) = \{83, 89, 97, 101, 103, 107, 113, 131, 139, 149, 151, 157, 173, 181, 191, 197, 199, 211, 223, 229, 271, 379, 487\}$$

$$M_p(7) = \{137, 167, 179, 193, 227, 233, 239, 241, 251, 263, 269, 277, 281, 283, 293, 307, 311, 313, 317, 331, 337, 347, 349, 367, 373, 383, 397, 419, 421, 431, 433, 439, 457, 463, 491, 509, 523, 541, 547, 571, 631, 653, 757, 811, 883, 1459\}$$

$$M_p(8) = \{257, 353, 359, 389, 401, 409, 443, 449, 461, 467, 479, 499, 503, 521, 557, 563, 569, 577, 587, 593, 599, 601, 607, 613, 617, 619, 643, 647, 659, 661, 673, 677, 683, 691, 701, 709, 727, 733, 739, 743, 751, 761, 787, 797, 827, 829, 839, 859, 859, 863, 877, 907, 911, 919, 937, 947, 967, 983, 991, 1009, 1019, 1033, 1039, 1051, 1063, 1087, 1091, 1093, 1103, 1117, 1171, 1279, 1291, 1297, 1303, 1307, 1373, 1423, 1471, 1483, 1549, 1567, 1597, 1621, 1627, 1783, 1949, 1999, 2053, 2269, 2287, 2647, 2917, 3079\}$$

$$\begin{aligned}
M_p(9) = & \{ 641, 719, 769, 773, 809, 821, 823, 857, 881, 887, \\
& 929, 941, 953, 971, 977, 997, 1013, 1021, 1031, 1049, \\
& 1061, 1069, 1109, 1123, 1129, 1151, 1153, 1163, 1181, 1187, \\
& 1193, 1201, 1213, 1217, 1223, 1229, 1231, 1237, 1249, 1259, \\
& 1277, 1289, 1301, 1319, 1321, 1327, 1367, 1381, 1399, 1427, \\
& 1429, 1447, 1451, 1453, 1481, 1487, 1489, 1493, 1499, 1511, \\
& 1523, 1531, 1559, 1671, 1579, 1583, 1697, 1609, 1613, 1657, \\
& 1663, 1667, 1669, 1693, 1699, 1709, 1721, 1723, 1733, 1741, \\
& 1747, 1753, 1759, 1777, 1787, 1789, 1801, 1811, 1823, 1831, \\
& 1847, 1861, 1867, 1873, 1877, 1879, 1901, 1933, 1851, 1979, \\
& 1987, 2003, 2011, 2017, 2019, 2039, 2083, 2087, 2089, 2111, \\
& 2129, 2131, 2143, 2161, 2179, 2203, 2207, 2213, 2221, 2237, \\
& 2239, 2243, 2251, 2281, 2293, 2311, 2341, 2357, 2371, 2377, \\
& 2383, 2389, 2399, 2423, 2437, 2503, 2521, 2539, 2549, 2557, \\
& 2591, 2593, 2609, 2617, 2677, 2683, 2699, 2711, 2719, 2731, \\
& 2749, 2791, 2843, 2851, 2887, 2927, 2971, 3011, 3049, 3067, \\
& 3109, 3187, 3253, 3259, 3271, 3307, 3319, 3331, 3511, 3529, \\
& 3533, 3547, 3557, 3583, 3613, 3727, 3823, 3889, 3907, 3919, \\
& 3943, 4051, 4159, 4219, 4447, 4549, 4663, 4789, 4861, 4871, \\
& 4903, 5023, 5347, 5419, 5869, 6823, 6967, 8803 \} \\
M_p(10) = & \{ 1097, 1283, 1361, 1409, 1433, 1439, 1543, 1553, 1601, 1619, \\
& 1637, 1697, 1871, 1889, 1907, 1913, 1931, 1973, 1993, 1997, \\
& 2027, 2063, 2069, 2081, 2099, 2113, 2137, 2141, 2153, 2257, \\
& 2273, 2297, 2309, 2333, 2339, 2347, 2351, 2381, 2393, 2411, \\
& 2417, 2441, 2447, 2459, 2467, 2473, 2477, 2531, 2543, 2551, \\
& 2579, 2621, 2633, 2659, 2663, 2671, 2687, 2689, 2693, 2707, \\
& 2713, 2729, 2753, 2767, 2777, 2797, 2801, 2803, 2833, 2837, \\
& 2857, 2861, 2897, 2903, 2909, 2939, 2953, 2957, 2963, 2969, \\
& 2999, 3001, 3019, 3023, 3037, 3041, 3061, 3083, 3119, 3121, \\
& 3137, 3163, 3167, 3169, 3181, 3191, \text{ to be continued.... } \}
\end{aligned}$$

また, Euler による無限和と無限積の恒等式を真似れば次の恒等式は明らかであろう。

$$\begin{aligned} \sum_{k=0}^{\infty} m(k)x^k &= \prod_{k=1}^{\infty} \prod_{i=1}^{c_k} \frac{1}{1-x^k} = \prod_{k=1}^{\infty} \frac{1}{(1-x^k)^{c_k}} \\ &= \prod_p \frac{1}{1-x^{L(p)}} \end{aligned}$$

$m(n)$ までを求めたければ

$$\prod_{k=1}^n \frac{1}{(1-x^k)^{c_k}}$$

の計算でよい。

例えば, $c_1 = 2, c_2 = 2, c_3 = 3, c_4 = 6, c_5 = 12, c_6 = 23, c_7 = 46, c_8 = 94, c_9 = 198, \dots$ より

$$\begin{aligned} \prod_{k=1}^9 \frac{1}{(1-x^k)^{c_k}} &= \frac{1}{(1-x)^2} \cdot \frac{1}{(1-x^2)^2} \cdot \frac{1}{(1-x^3)^3} \cdot \frac{1}{(1-x^3)^3} \cdot \frac{1}{(1-x^4)^6} \cdot \\ &\quad \frac{1}{(1-x^5)^{12}} \cdot \frac{1}{(1-x^6)^{23}} \cdot \frac{1}{(1-x^7)^{46}} \cdot \frac{1}{(1-x^8)^{94}} \cdot \frac{1}{(1-x^9)^{198}} \\ &= 1 + 2x + 5x^2 + 11x^3 + 26x^4 + 59x^5 + 137x^6 + 312x^7 + 719x^8 + 1651x^9 + \dots \end{aligned}$$

$\therefore m(1) = 2, m(2) = 5, m(3) = 11, m(4) = 26, m(5) = 59, m(6) = 137, m(7) = 312, m(8) = 719, m(9) = 1651, \dots$

$c_n m(n)$ について (by Python)

n	c_n	$m(n)$
1	2	2
2	2	5
3	3	11
4	6	26
5	12	59
6	23	137
7	46	312
8	94	719
9	198	1651
10	424	3816
11	854	8757
12	1859	20202
13	3884	46440
14	8362	106957
15	17837	245989

Python による計算プログラム

```
LIMIT = 15
N = 3**LIMIT + 1
phi = [i for i in range(N)]
for i in range(2, N):
    if phi[i] == i:
        for j in range(i, N, i):
            phi[j] -= phi[j] // i

L = [0] * N
for i in range(2, N):
    if i % 2 == 0:
        L[i] = L[phi[i]] + 1
    else:
        L[i] = L[phi[i]]

c = [0] * (LIMIT + 1)
m = [0] * (LIMIT + 1)
for i in range(1, N):
    x = L[i]
    if x <= LIMIT:
        m[x] += 1
        if phi[i] == i - 1:
            c[x] += 1

print("n, c_n, m(n)")
for n in range(LIMIT + 1):
    print(n, c[n], m[n], sep=", ")
```

また、 $M_p(n)$ の要素 q については

$$\max_{q \in M_p(n)} q \leq 2 \cdot 3^{n-1} + 1$$

が成立しているが、この不等式での等号は外せない。 $2 \cdot 3^{n-1} + 1$ が素数になる場合があるからである。ただし、 $n \rightarrow \infty$ では、(実験では) 素数になる確率は下がってゆくような振舞いを見せている・・・。

因みに $n < 5000$ の範囲で $y(n) = 2 \cdot 3^{n-1} + 1$ が素数となるのは $n = 1, 2, 3, 5, 6, 7, 10, 17, 18, 31, 55, 58, 61, 66, 133, 181, 321, 697, 783, 823, 898, 1253, 1455, 4128$ の 24 個だけである (by Maple & Python)。

```
y(181) = 1523546 9609173278 4678579455 4412311235 0084960280 4790393448 0031314899
1427468606 6076039203
```

```
def prime_test(p):
    q = p - 1
    p2 = q // 2
    p3 = q // 3
    for a in (2, 3, 5, 6, 7):
```

```
    if pow(a, p2, p) != q:  
        continue  
    if pow(a, p3, p) != 1:  
        return a  
return -1
```

```
c = 0  
for n in range(3, 5000):  
    # if n % 100 == 0: print("****", n)  
    if n % 4 == 0: continue  
    if n % 5 == 4: continue  
    if n % 6 == 2: continue  
    yn = 2 * 3**(n-1) + 1  
    x = pow(2, yn - 1, yn)  
    if x != 1: continue  
    root = prime_test(yn)  
    c += 1  
    print(c, n, root)
```