

Project Euler 35. Circular Primes

hiragn

2024 年 12 月 21 日

1. 問題の概要

197 は巡回素数と呼ばれる。桁を回転させたときに得られる数 197, 971, 719 がすべて素数だからである。100 未満の巡回素数は 13 個ある。

2, 3, 5, 7, 11, 13, 17, 31, 37, 71, 73, 79, 97

100 万未満の巡回素数はいくつあるか？

<https://projecteuler.net/problem=35>

2. RotateLeft で回転

「桁の回転」は `IntegerDigits` で作った桁数字のリストを `RotateLeft` して `FromDigits` で数に戻せば実現できます。

もとの数から回転で移れる数のリストを作って、それらがすべて素数かどうか調べました。対象となる素数は `PrimePi[106]=78,498` 個です。

```

1 In[] := Clear["Global`*"];
2 RepeatedTiming[
3   cond[n_] :=
4     AllTrue[FromDigits@RotateLeft[IntegerDigits@n, #] & /@
5       Range@IntegerLength@n, PrimeQ];
6   ans = Length@Parallelize@Select[Prime@Range@PrimePi[10^6], cond]]
7
8 Out[] = {0.37419, 55}

```

3. グラフのサイクルを数える

素数 i から 1 回の回転で素数 j に移れるとき i から j に辺を張ってグラフを作ります。このグラフのサイクルに属する頂点の個数が答え。

回転で自分自身に移る 2, 3, 5, 7, 11 は別枠として数えると解けます。こちらの方が速いのは意外でした。

```

1 In[] := Clear["Global`*"];
2 RepeatedTiming[
3   nmax = 10^6;
4   f[n_] := Module[{m = FromDigits@RotateLeft@IntegerDigits@n},
5     If[PrimeQ@m, n -> m, Nothing]];
6   g = Graph[f /@ Prime@Range@PrimePi@nmax];
7   ans = Length@Union[{2, 3, 5, 7, 11},
8     Flatten[VertexList /@ FindCycle[g, Infinity, All]]]]
9
10 Out[] = {0.303139, 55}

```
