

On the Computation of Primes

By [Clive Jones](#)

Last Revised: 2018 Feb 3

Abstract: "Featuring the PF5 Prime-Generating Function"

- 1> Primes may be created with a '*Primes From 5*' formula:
- 2> $PF5(N) = (N*3) + 1$ if N is even
- 3> $+ 2$ if N is odd For $N > 0$ & Integer
- 4> Outputs both Primes & Semi-Primes.
- 5> Primes arise from 'P' input-values.
- 6> Semi-Primes arise from 'SP' input-values.

| Prime Output | Semi-Prime Output |
|--|--|
| P input values: | SP input values: |
| 1, 2, 3, 4, 5, 6, 7, 9, 10, 12, 13, 14, 15, 17, 19, 20, 22-24, 26, 27, 29, 32-37, 42, 43, 45, 46, 49, 50, 52, 54, 55, 57, 59, 60, 63-66, 70, 74-77, 79, 80, 83, 85, 87, 89, 90, 92-94, 97, 102-105 ... | 8, 11, 16, 18, 21, 25, 28, 30, 31, 38, 39, 40, 41, 44, 47, 48, 51, 53, 56, 58, 61, 62, 67, 68, 69, 71-73, 78, 81, 82, 84, 86, 88, 91, 95, 96, 98-101 ... |
| (60% of the time for $N < 101$) | (40% of the time for $N < 101$) |
| Eg: $5*3+2 = 17$ (Prime) $12*3+1 = 37$ (Prime) | Eg: $11*3+2=35$ (Semi-Prime) $16*3+1=49$ (Semi-Prime) |

Fig 1: Low-Order PF5 Inputs & Corresponding Output Type.

- 7> SP input values show a pattern & can be predicted.
- 8> Avoiding SP input values produces (desired) prime-only output.

- 9> Whilst not able to provide the Nth prime directly, the PF5 function should yield all the primes up to and including the Nth prime in a fairly rapid manner, software permitting.
- 10> The SP inputs conform to lists which are initiated by a given root and which contain predictable even/odd values which produce multiples of the root as their Semi-Prime output.

| Root | Square | N | P+Q | Interval | SP Input |
|------|--------|-----|-------|----------|--|
| 5 | 25 | 8 | 3+7 | 10 | 8, 11, 18, 21, 28, 31, 38, 41, 48, 51... |
| 7 | 49 | 16 | 9+5 | 14 | 16, 25, 30, 39, 44, 53, 58, 67, 72... |
| 11 | 121 | 40 | 7+15 | 22 | 40, 47, 62, 69, 84, 91, 136... |
| 13 | 169 | 56 | 17+9 | 26 | 56, 73, 82, 99... |
| 17 | 289 | 96 | 11+23 | 34 | 96, 107, 130, 141... |
| 19 | 361 | 120 | 25+13 | 38 | 120, 145, 158, 183... |
| 23 | 529 | 176 | 15+31 | 46 | 176, 191, 222, 237... |
| 29 | 841 | 280 | 33+17 | 50 | 280, 313, 330, 363... |
| 31 | 961 | 320 | 19+39 | 58 | 320, 339... |
| 37 | 1369 | 456 | 41+21 | 62 | 456, 497... |
| 41 | 1681 | 560 | 23+47 | 70 | 560, 583... |
| 43 | 1849 | 616 | 49+25 | 74 | 616, 665... |
| 47 | 2209 | 736 | 27+55 | 82 | 736, 763... |

Fig 2: PF5 List-Based Analysis.

- 11> The P+Q values are the magic-numbers that help computation.
- 12> Smallest P or Q increments by 2 and flips position.
- 13> Greatest P or Q increments by {2, 6} and flips position.
- 14> Interval First-Difference is {4, 8}
- 15> Interval between successive even/odd SP inputs is P+Q.
- 16> Adding P then Q gives next_odd then next_even term.
- 17> The software that builds the prime-list is still a bit glitchy at present but a 'pattern in the primes' or more accurately in the SP inputs to the PF5 function is seemingly present.