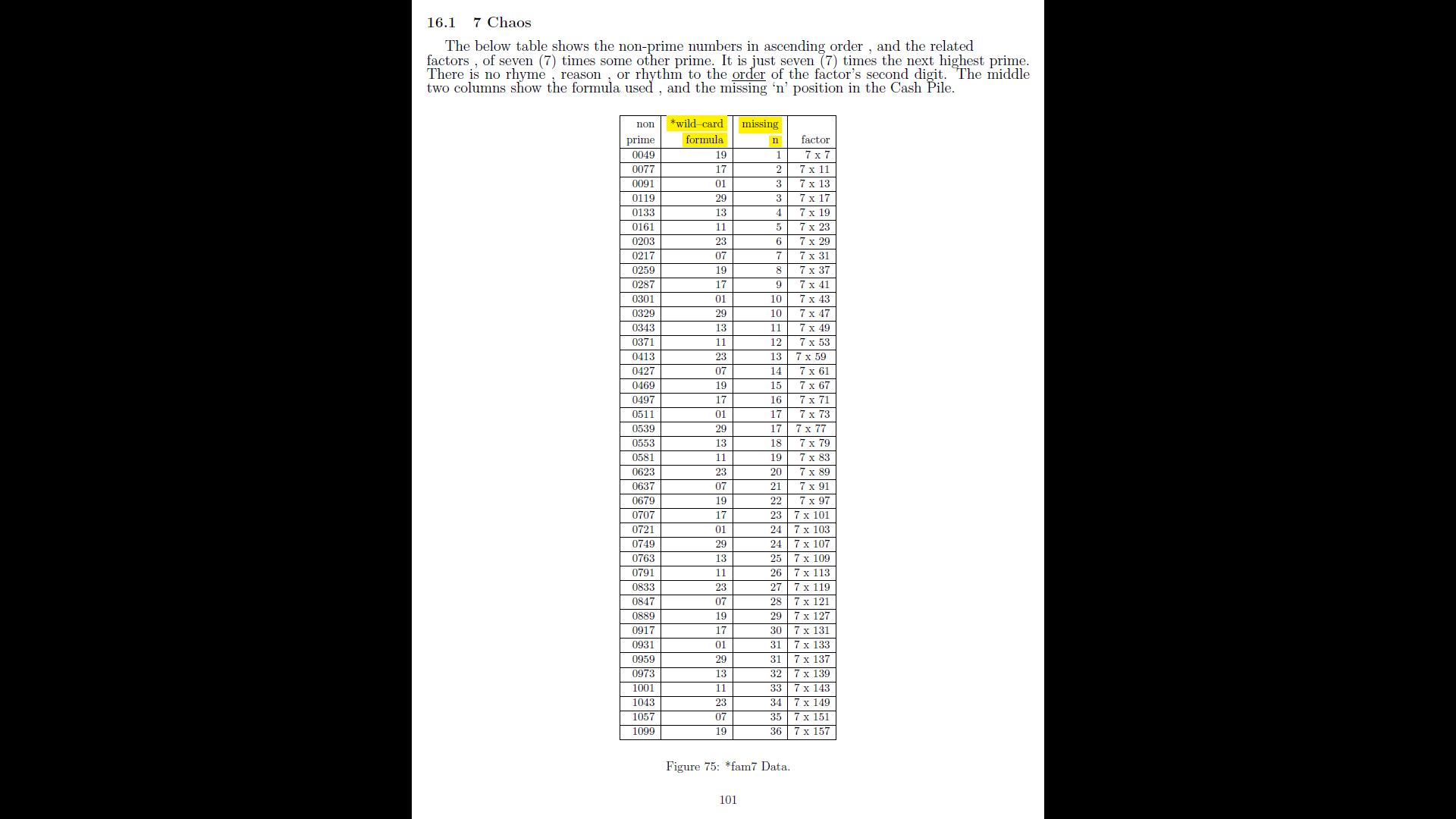
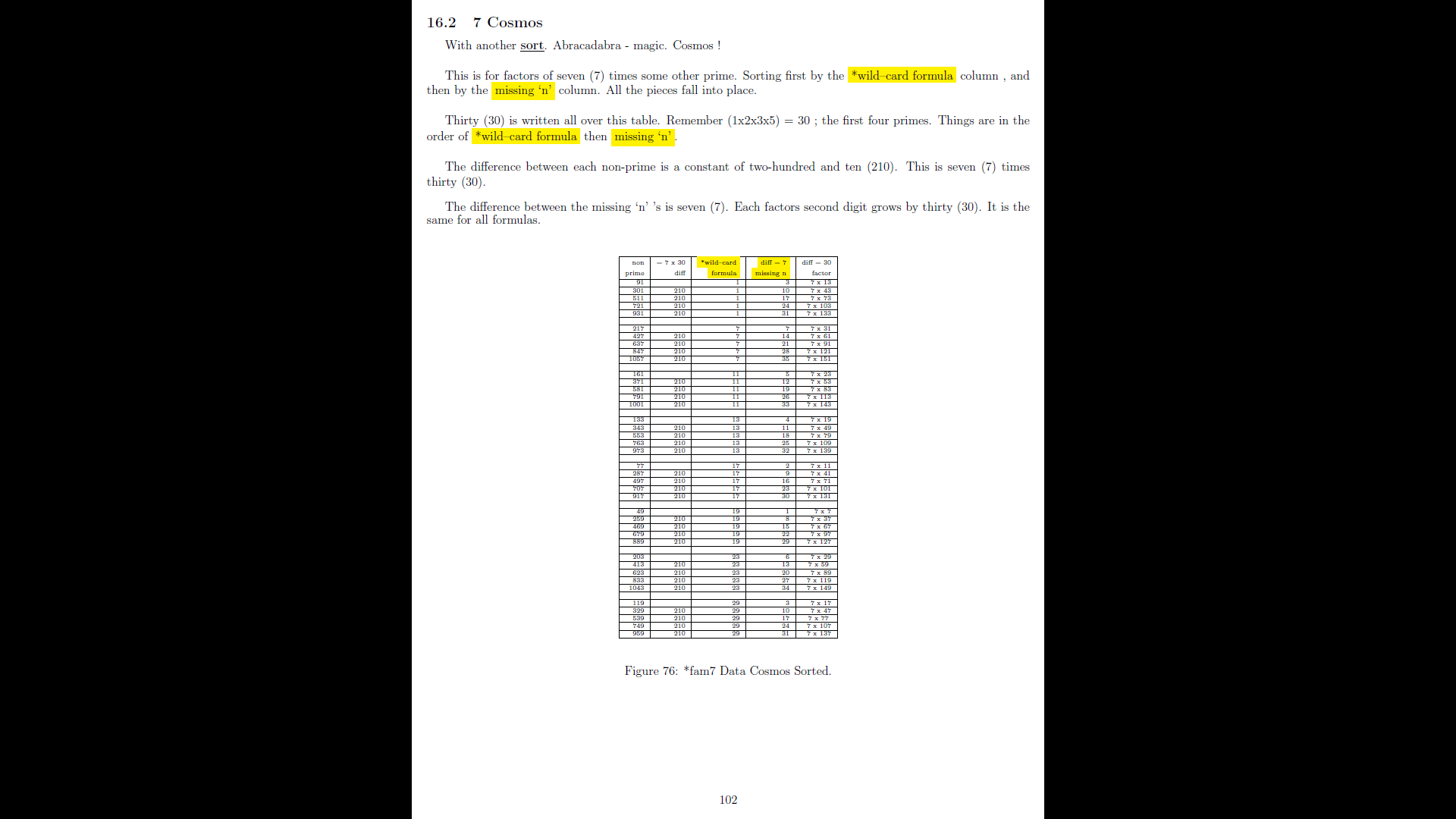
just another 30 and the keys

The solution to finding all non-prime numbers will sort out all the primes.

Using \*wildcard (\*1 , \*3 , \*7 , \*9) to create \*fam ( \*fam01 , \*fam07 , \*fam11 , \*fam13 , \*fam17 , \*fam19 , \*fam23 , and \*fam29 )a clear pattern of all numbers can be derived.

Early in the TioCash.pdf file there were chapters 16 , 17 , 18 – Chaos to Cosmos ; showed how a \*fam was separated by the \*fam and a constant of thirty (30).





16.1 7 Chaos

The below table shows the non-prime numbers in ascending order , and the related

factors , of seven (7) times some other prime. It is just seven (7) times the next highest prime.

There is no rhyme , reason , or rhythm to the order of the factor's second digit. The middle

two columns show the formula used , and the missing `n' position in the Cash Pile.

non-prime \*fam MN factor

0049 19 1 7 x 7

0077 17 2 7 x 11

0091 01 3 7 x 13

0119 29 3 7 x 17

0133 13 4 7 x 19

0161 11 5 7 x 23

0203 23 6 7 x 29

0217 07 7 7 x 31

0259 19 8 7 x 37

0287 17 9 7 x 41

0301 01 10 7 x 43

0329 29 10 7 x 47

0343 13 11 7 x 49

0371 11 12 7 x 53

0413 23 13 7 x 59

0427 07 14 7 x 61

0469 19 15 7 x 67

0497 17 16 7 x 71

0511 01 17 7 x 73

0539 29 17 7 x 77

0553 13 18 7 x 79

0581 11 19 7 x 83

0623 23 20 7 x 89

0637 07 21 7 x 91

0679 19 22 7 x 97

0707 17 23 7 x 101

0721 01 24 7 x 103

0749 29 24 7 x 107

0763 13 25 7 x 109

0791 11 26 7 x 113

0833 23 27 7 x 119

0847 07 28 7 x 121

0889 19 29 7 x 127

0917 17 30 7 x 131

0931 01 31 7 x 133

0959 29 31 7 x 137

0973 13 32 7 x 139

1001 11 33 7 x 143

1043 23 34 7 x 149

1057 07 35 7 x 151

1099 19 36 7 x 157

16.2 7 Cosmos

With another sort. Abracadabra - magic. Cosmos ! This is for factors of seven (7) times some other prime. Sorting first by the \*wild-card formula column , and then by the missing `n' column. All the pieces fall into place. Thirty (30) is written all over this table. Remember (1x2x3x5) = 30 ; the first four primes. Things are in the order of \*wild{card formula then missing `n' . The difference between each non-prime is a constant of two-hundred and ten (210). This is seven (7) times thirty (30). The difference between the missing `n' 's is seven (7). Each factors second digit grows by thirty (30). It is the same for all formulas.

non-prime 7x30 \*fam diff = 7 diff = 30

diff MN - missing n factors

91 1 3 7 x 13

301 210 1 10 7 x 43

511 210 1 17 7 x 73

721 210 1 24 7 x 103

931 210 1 31 7 x 133

217 7 7 7 x 31

427 210 7 14 7 x 61

637 210 7 21 7 x 91

847 210 7 28 7 x 121

1057 210 7 35 7 x 151

161 11 5 7 x 23

371 210 11 12 7 x 53

581 210 11 19 7 x 83

791 210 11 26 7 x 113

1001 210 11 33 7 x 143

133 13 4 7 x 19

343 210 13 11 7 x 49

553 210 13 18 7 x 79

763 210 13 25 7 x 109

973 210 13 32 7 x 139

77 17 2 7 x 11

287 210 17 9 7 x 41

497 210 17 16 7 x 71

707 210 17 23 7 x 101

917 210 17 30 7 x 131

49 19 1 7 x 7

259 210 19 8 7 x 37

469 210 19 15 7 x 67

679 210 19 22 7 x 97

889 210 19 29 7 x 127

203 23 6 7 x 29

413 210 23 13 7 x 59

623 210 23 20 7 x 89

833 210 23 27 7 x 119

1043 210 23 34 7 x 149

119 29 3 7 x 17

329 210 29 10 7 x 47

539 210 29 17 7 x 77

749 210 29 24 7 x 107

959 210 29 31 7 x 137

%‘just another thirty’

After separating a PS (Prime Suspect) by its \*fam , several additional rules can be applied.

1. \*fam ( \*fam01 , \*fam07 , \*fam11 , \*fam13 , \*fam17 , \*fam19 , \*fam23 , and \*fam29 ) will generate all non-prime numbers.
2. For each \*fam the next PS is plus 30.

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  | \*fam01 | \*fam07 | \*fam11 | \*fam13 | \*fam17 | \*fam19 | \*fam23 | \*fam29 |
| prime | 1 | 7 | 11 | 13 | 17 | 19 | 23 | 29 |
| PS + 30 | 31 | 37 | 41 | 43 | 47 | 49 | 53 | 59 |
|  | 61 | 67 | 71 | 73 | 77 | 79 | 83 | 89 |
|  | 91 | 97 | 101 | 103 | 107 | 109 | 113 | 119 |
|  | 121 | 127 | 131 | 133 | 137 | 139 | 143 | 149 |
|  | 151 | 157 | 161 | 163 | 167 | 169 | 173 | 179 |

1. Take each \*fam and subtract from PS.
2. Then divide by 30 to get the level (PSL Prime Suspect Level). This is the height of the PS in the Cash Pile.
   1. Look at 7: (7-\*fam07)/30 = PSL (level) 0
   2. Look at 37: (37-\*fam07)/30 = PSL (level) 1
   3. Look at 67: (67-\*fam07)/30 = PSL (level) 2
   4. Just another 30 . . .
3. Take the PS and find its square root ( √PS).
4. Since the values are fixed distance apart (just another 30) , there are now sub-keys for each \*fam. The two sub-keys are constant a difference of 30 , and the PSL (level) will grow by the \*fam.
5. Inside each \*fam is a unique set of sub-keys. That can not be applied to any other \*fam.
6. These sub-keys will show all non-prime numbers.
7. These sub-keys start from the cosmos sorts.
8. In the following tables are the sub-keys by \*fam. Note that \*fam is does not create a non-prime , then PS is prime.

|  |  |  |
| --- | --- | --- |
| **\*fam07** |  |  |
| \*fam7 | 1 and +30 | L0 and +7 |
| \*fam11 | 17 and +30 | L6 and + 11 |
| \*fam13 | 19 and + 30 | L21 an + 13 |
| \*fam17 | 11 and + 30 | L23 and + 17 |
| \*fam19 | 13 and + 30 | L8 and +19 |
| \*fam23 | 29 and + 30 | L22 and + 23 |
| \*fam29 | 23 and + 30 | L22 and + 29 |

|  |  |  |
| --- | --- | --- |
| **\*fam11** |  |  |
| \*fam7 | 23 and +30 | L5 and +7 |
| \*fam11 | 1 and +30 | L0 and + 11 |
| \*fam13 | 17 and + 30 | L7 an + 13 |
| \*fam17 | 13 and + 30 | L7 and + 17 |
| \*fam19 | 29 and + 30 | L18 and +19 |
| \*fam23 | 7 and + 30 | L5 and + 23 |
| \*fam29 | 19 and + 30 | L18 and + 29 |

|  |  |  |
| --- | --- | --- |
| **\*fam13** |  |  |
| \*fam7 | 19 and +30 | L4 and +7 |
| \*fam11 | 23 and +30 | L8 and + 11 |
| \*fam13 | 1 and + 30 | L0 an + 13 |
| \*fam17 | 29 and + 30 | L16 and + 17 |
| \*fam19 | 7 and + 30 | L4 and +19 |
| \*fam23 | 11 and + 30 | L8 and + 23 |
| \*fam29 | 17 and + 30 | L16 and + 29 |

|  |  |  |
| --- | --- | --- |
| **\*fam17** |  |  |
| \*fam7 | 11 and +30 | L2 and +7 |
| \*fam11 | 7 and +30 | L2 and + 11 |
| \*fam13 | 29 and + 30 | L12 and + 13 |
| \*fam17 | 1 and + 30 | L0 and + 17 |
| \*fam19 | 23 and + 30 | L14 and +19 |
| \*fam23 | 19 and + 30 | L14 and + 23 |
| \*fam29 | 13 and + 30 | L12 and + 29 |

|  |  |  |
| --- | --- | --- |
| **\*fam19** |  |  |
| \*fam7 | 7 and +30 | L1 and +7 |
| \*fam11 | 29 and +30 | L10 and + 11 |
| \*fam13 | 13 and + 30 | L5 and + 13 |
| \*fam17 | 17 and + 30 | L9 and + 17 |
| \*fam19 | 1 and + 30 | L0 and +19 |
| \*fam23 | 23 and + 30 | L17 and + 23 |
| \*fam29 | 11 and + 30 | L10 and + 29 |

|  |  |  |
| --- | --- | --- |
| **\*fam23** |  |  |
| \*fam7 | 29 and +30 | L6 and +7 |
| \*fam11 | 13 and +30 | L4 and + 11 |
| \*fam13 | 11 and + 30 | L4 and + 13 |
| \*fam17 | 19 and + 30 | L10 and + 17 |
| \*fam19 | 17 and + 30 | L10 and +19 |
| \*fam23 | 1 and + 30 | L0 and + 23 |
| \*fam29 | 7 and + 30 | L6 and + 29 |

|  |  |  |
| --- | --- | --- |
| **\*fam29** |  |  |
| \*fam7 | 17 and +30 | L3 and +7 |
| \*fam11 | 19 and +30 | L6 and + 11 |
| \*fam13 | 23 and + 30 | L9 and + 13 |
| \*fam17 | 7 and + 30 | L3 and + 17 |
| \*fam19 | 11 and + 30 | L6 and +19 |
| \*fam23 | 13 and + 30 | L9 and + 23 |
| \*fam29 | 1 and + 30 | L0 and + 29 |

1. Here is a comparison between \*fam13 and \*fam17 – a pair the adds up to ‘just another 30’. Here are the sub-keys.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **\*fam13** |  |  | **\*fam17** |  |
| \*fam7 | 19 and +30 |  | \*fam7 | 11 and +30 |
| \*fam11 | 23 and +30 |  | \*fam11 | 7 and +30 |
| \*fam13 | 1 and + 30 |  | \*fam13 | 29 and + 30 |
| \*fam17 | 29 and + 30 |  | \*fam17 | 1 and + 30 |
| \*fam19 | 7 and + 30 |  | \*fam19 | 23 and + 30 |
| \*fam23 | 11 and + 30 |  | \*fam23 | 19 and + 30 |
| \*fam29 | 17 and + 30 |  | \*fam29 | 13 and + 30 |

1. \*fam7 = 19 + 11 = 30
2. \*fam11 = 23 + 7 = 30
3. \*fam13 = 1 + 29 = 30 (note 1 is with 13 for \*fam13) same for each \*fam.
4. \*fam17 = 29 + 1 = 30 (note 1 is with 17 for \*fam17)
5. \*fam19 = 7 + 23 = 30
6. \*fam23 = 11 + 19 = 30
7. \*fam29 = 17 + 13 = 30
8. Only these sub-keys will satisfy a PS at this level. We have sorted off all the other possible solutions.
9. Loop thru the \*fam and add sub-keys ; 30 and the level offset.
10. Test if the sub-key creates a modulo zero.
11. Loop until the sq root of PS.

Think in these terms , we are taking a slice of numbers from 0 to 29 , test the PS to see which \*wildcard , then which \*fam. Work thru the found \*fam and loop thru until the sq root ; using the sub-keys. If test with mod 1 for 0 , then composite number. If by the time we hit the sq root – then prime.

Since we have slowly eliminated number by sorting , and more sorting all that can be left is prime. This is a backdoor approach , find and eliminate all non-prime numbers and the last sort is \*fam1. If there are no other solutions to the PS in the loop , then the default is \*fam1 = prime.

Look at \*fam07 sub-keys and a table of data.

Look in file: “look late 009 just another 30 fam07 keys.xlsx”



Using the sub-keys for the \*fam07 there are only eight solutions.



This is all that is left to solve for a non-prime number. The top box shows the difference between level (PSL) zero (0) and next run +30 and +\*fam in the top title. This is the next factor.

The bottom box shows the original sub-keys for this \*fam07. While looping to the sq root of PS , if one of these sub-keys solves for modulo zero , then non-prime.

After completion only prime is left.

Back to the big table. Here are the headings:

|  |
| --- |
| ps |
| 7 |

This is the beginning of \*fam07 – prime the well for future numbers. All future PS in this \*fam are ‘just another 30 away’.



This is a marker , just to show what number we are working with. It is a sanity check – where am I , who am I ?



The sq root and the next higher integer value.



This is the level (PSL) in the Cash Pile , it is the result of the following:

Here for \*fam07

=(PS-07)/30

Generically it is:

=(PS-\*fam)/30



The unique sub-keys for \*fam07.

Here is the last sort , some might call it a filter table. \*fam1 = the PS. It is the beginning starting at 7 and adding +30 to each next row. The rest of the headings are the \*fam ’s from seven to twenty-nine. The “by #’ title column is being divided into the PS (\*fam1).

If the result is a number that is equal to zero for modulo 1 , then it has a factor.

Look at the ‘Factor’ title , this is a simple TRUE of FALSE if there is a mod 0.

TRUE is a composite number (factors)

FALSE is a prime number.

The loop and sub keys:



The first row is the sub-key \*fam for \*fam07.

The second row is the ‘prime the pump’ starting number for the sub-key. Each next factor is +30 ; the second row in above table.

The third row is the ‘prime the pump’ starting level and the plus delta to the next level. And , each next level (PSL) +\*fam ; the third row.

Fall thru if PS falls thru these sub-keys , it is prime. Note the FALSE rows in the big table , the sub-keys can not solve for mod 0 ; what is left is prime.