One observation about primes

Nikolay Dementev

3-192 Kholzunova Street, Voronezh, 394068, Russia

Based on the observation of randomly chosen primes it has been conjectured that the sum of digits that form any prime number should yield either even number or another prime number. The conjecture was successfully tested for the first 100 primes.

KEYWORDS: primes, prime numbers

1. The Observation

Based on the observation of randomly chosen primes, a following statement has been conjectured:

For any multi-digital prime number (presented in decimal number system) the sum of its digits yields either even number or another prime number. (1)

The statement (1) was successfully tested for the first 100 primes (see Table 1) as they are listed elsewhere [1]. To test the converse:

Any odd number (presented in decimal number system and not divisible by 5), which sum of digits yields a prime (except 2 or 3), is a prime itself. (2)

the program in BASIC (see Appendix) was run. The first 100 numbers being generated based on the statement (2) are presented in Table 2. As one can see, even though the statement (2) is *mostly* correct, the statement (2) is wrong in general.

2. Concluding Remarks

a) It would be interesting to try proving the conjecture (1) and to see whether it holds for other number systems or not.

b) Although statement (2) is wrong in general and for this reason cannot be used as a leading principle in prime numbers generator, a slightly modified statement (1) (see below as statement (3)) can be used to sieve out the odd numbers which are not primes:

Any multi-digital odd number (presented in decimal number system) is not a prime number if the sum of its digits neither an even number nor a prime. (3)

References

1. http://primes.utm.edu/lists/small/1000.txt

Table 1.

Prime		SUM of digits		Prime		SUM of digits		Prime		SUM of digits	
P(1)	2			P(34)	139	13	P(6)	P(67)	331	7	P(4)
P(2)	3			P(35)	149	14	even	P(68)	337	13	P(6)
P(3)	5			P(36)	151	7	P(4)	P(69)	347	14	even
P(4)	7			P(37)	157	13	P(6)	P(70)	349	16	even
P(5)	11	2	even	P(38)	163	10	even	P(71)	353	11	P(5)
P(6)	13	4	even	P(39)	167	14	even	P(72)	359	17	P(7)
P(7)	17	8	even	P(40)	173	11	P(5)	P(73)	367	16	even
P(8)	19	10	even	P(41)	179	17	P(7)	P(74)	373	13	P(6)
P(9)	23	5	P(3)	P(42)	181	10	even	P(75)	379	19	P(8)
P(10)	29	11	P(5)	P(43)	191	11	P(5)	P(76)	383	14	even
P(11)	31	4	even	P(44)	193	13	P(6)	P(77)	389	20	even
P(12)	37	10	even	P(45)	197	17	P(7)	P(78)	397	19	P(8)
P(13)	41	5	P(3)	P(46)	199	19	P(8)	P(79)	401	5	P(3)
P(14)	43	7	P(4)	P(47)	211	4	even	P(80)	409	13	P(6)
P(15)	47	11	P(5)	P(48)	223	7	P(4)	P(81)	419	14	even
P(16)	53	8	even	P(49)	227	11	P(5)	P(82)	421	7	P(4)
P(17)	59	14	even	P(50)	229	13	P(6)	P(83)	431	8	even
P(18)	61	7	P(4)	P(51)	233	8	even	P(84)	433	10	even
P(19)	67	13	P(6)	P(52)	239	14	even	P(85)	439	16	even
P(20)	71	8	even	P(53)	241	7	P(4)	P(86)	443	11	P(5)
P(21)	73	10	even	P(54)	251	8	even	P(87)	449	17	P(7)
P(22)	79	16	even	P(55)	257	14	even	P(88)	457	16	even
P(23)	83	11	P(5)	P(56)	263	11	P(5)	P(89)	461	11	P(5)
P(24)	89	17	P(7)	P(57)	269	17	P(7)	P(90)	463	13	P(6)
P(25)	97	16	even	P(58)	271	10	even	P(91)	467	17	P(7)
P(26)	101	2	even	P(59)	277	16	even	P(92)	479	20	even
P(27)	103	4	even	P(60)	281	11	P(5)	P(93)	487	19	P(8)
P(28)	107	8	even	P(61)	283	13	P(6)	P(94)	491	14	even
P(29)	109	10	even	P(62)	293	14	even	P(95)	499	22	even
P(30)	113	5	P(3)	P(63)	307	10	even	P(96)	503	8	even
P(31)	127	10	even	P(64)	311	5	P(3)	P(97)	509	14	even
P(32)	131	5	P(3)	P(65)	313	7	P(4)	P(98)	521	8	even
P(33)	137	11	P(5)	P(66)	317	11	P(5)	P(99)	523	10	even
								P(100)	541	10	even

Table 2.

Generated numbers G(…)			**Generated numbers G(…)**			**Generated numbers G(…)**		
G(1)	7	prime	G(34)	263	prime	G(67)	487	prime
G(2)	23	prime	G(35)	269	prime	**G(68)**	**511**	**NOT prime**
G(3)	29	prime	G(36)	281	prime	**G(69)**	**517**	**NOT prime**
G(4)	41	prime	G(37)	283	prime	**G(70)**	**533**	**NOT prime**
G(5)	43	prime	**G(38)**	**287**	**NOT prime**	**G(71)**	**539**	**NOT prime**
G(6)	47	prime	**G(39)**	**289**	**NOT prime**	**G(72)**	**551**	**NOT prime**
G(7)	**49**	**NOT prime**	G(40)	311	prime	**G(73)**	**553**	**NOT prime**
G(8)	61	prime	G(41)	313	prime	G(74)	557	prime
G(9)	67	prime	G(42)	317	prime	**G(75)**	**559**	**NOT prime**
G(10)	83	prime	**G(43)**	**319**	**NOT prime**	G(76)	571	prime
G(11)	89	prime	G(44)	331	prime	G(77)	577	prime
G(12)	113	prime	G(45)	337	prime	G(78)	593	prime
G(13)	**119**	**NOT prime**	G(46)	353	prime	G(79)	599	prime
G(14)	131	prime	G(47)	359	prime	G(80)	601	prime
G(15)	**133**	**NOT prime**	**G(48)**	**371**	**NOT prime**	G(81)	607	prime
G(16)	137	prime	G(49)	373	prime	**G(82)**	**623**	**NOT prime**
G(17)	139	prime	**G(50)**	**377**	**NOT prime**	**G(83)**	**629**	**NOT prime**
G(18)	151	prime	G(51)	379	prime	G(84)	641	prime
G(19)	157	prime	**G(52)**	**391**	**NOT prime**	G(85)	643	prime
G(20)	173	prime	G(53)	397	prime	G(86)	647	prime
G(21)	179	prime	G(54)	401	prime	**G(87)**	**649**	**NOT prime**
G(22)	191	prime	**G(55)**	**403**	**NOT prime**	G(88)	661	prime
G(23)	193	prime	**G(56)**	**407**	**NOT prime**	**G(89)**	**667**	**NOT prime**
G(24)	197	prime	G(57)	409	prime	G(90)	683	prime
G(25)	199	prime	G(58)	421	prime	**G(91)**	**689**	**NOT prime**
G(26)	**203**	**NOT prime**	**G(59)**	**427**	**NOT prime**	**G(92)**	**713**	**NOT prime**
G(27)	**209**	**NOT prime**	G(60)	443	prime	G(93)	719	prime
G(28)	**221**	**NOT prime**	G(61)	449	prime	**G(94)**	**731**	**NOT prime**
G(29)	223	prime	G(62)	461	prime	G(95)	733	prime
G(30)	227	prime	G(63)	463	prime	G(96)	737	prime
G(31)	229	prime	G(64)	467	prime	G(97)	739	prime
G(32)	241	prime	**G(65)**	**469**	**NOT prime**	G(98)	751	prime
G(33)	**247**	**NOT prime**	**G(66)**	**481**	**NOT prime**	G(99)	757	prime
						G(100)	773	prime

Appendix

```
print "The program generates odd numbers G(...) not bigger than 9999"
print
print "G-numbers need to be checked whether they are primes or not"
print
input "Press [Enter] to continue."; dummyVariable
cls
rem 'Running through all the odd numbers. Each number is represented as (d c b a) where d, c, b, a - are its digits
rem 'for instance, number 3597 will be represented with d=3, c=5, b=9, a=7.
rem 'Because any number under consideration should be:
rem '1. odd number, 'a' should not be equal to 0,2,4,6,8
rem '2. not divisible by 5, 'a' should not be equal to 5
rem '3. not bigger than 9999, SUM=d+c+b+a will not be bigger than 9+9+9+9=36. Thus, the biggest prime number to check the SUM with is 31.
let counter=1
for d=0 to 9
   for c=0 to 9
      for b=0 to 9
         for a=1 to 9
         if a=2 then goto [NEXT]
         if a=4 then goto [NEXT]
         if a=5 then goto [NEXT]
         if a=6 then goto [NEXT]
         if a=8 then goto [NEXT]
            let SUM=d+c+b+a
            if SUM=5 then goto [PRINT]
            if SUM=7 then goto [PRINT]
            if SUM=11 then goto [PRINT]
            if SUM=13 then goto [PRINT]
            if SUM=17 then goto [PRINT]
            if SUM=19 then goto [PRINT]
            if SUM=23 then goto [PRINT]
            if SUM=29 then goto [PRINT]
            if SUM=31 then goto [PRINT]
            goto [NEXT]
            [PRINT]
            print "G(";counter;")="; d; c; b; a,"SUM=";SUM
            print
            counter=counter+1
         [NEXT]
         next a
      next b
   next c
next d
```