Annales Univ. Sci. Budapest., Sect. Comp. 32 (2010) 129-142

PRIMALITY TESTING USING GENERALIZED FIBONACCI SEQUENCES

A. Krem (Budapest, Hungary)

Abstract. The p-th element of a generalized Fibonacci sequence modulo p is computed. If p is a prime, the result is $p^{(q-1)/2} \bmod q$, where q is a prime of the form $4k+1$ and $q < p$, used here as a parameter. We have similar result for the $(p+1)$-th element and the other q prime values, too. The operations needed are approximately $3(\log_2 p)$ multiplications with mod p reductions and some additions. Very few not primes satisfy the test conditions (pseudoprimes). The number of pseudoprimes is considerably less than the number of the Carmichael numbers.

Let the sequence c_n be defined by

$$(a-b)c_n = a^n - b^n \qquad n = 1, 2, 3, \ldots,$$

where a and b are the roots of the next equation with integer coefficients:

$$x^2 - Px + Q = 0.$$

The coefficients are functions of the prime $q > 0$

$$P = 1 \quad \text{and} \quad Q = (1-q)/4, \quad \text{if} \quad q \equiv 1 \mod 4,$$

and

$$P = 2 \quad \text{and} \quad Q = 1-q, \quad \text{if} \quad q \equiv 2 \text{ or } q \equiv 3 \mod 4.$$

We will consider the two cases separately.

1. $q \equiv 1 \mod 4$

Now $P \equiv 1$ and $Q = (1-q)/4$. Thus a and b are the roots of the equation

$$x^2 - x + (1-q)/4 = 0,$$

namely

$$a = (1+\sqrt{q})/2 \quad \text{and} \quad b = (1-\sqrt{q})/2,$$

while $q \equiv 1 \mod 4$.

Obviously for the powers of a and b and their linear functions holds the recursion

$$c_{n+1} = c_n + c_{n-1}(q-1)/4.$$

One can easily prove that

$$c_{n+m} = c_{m+1}c_n + (c_{m+2} - c_{m+1})c_{n-1},$$

consequently

$$c_{2n} = c_n(2c_{n+1} - c_n) \quad \text{and} \quad c_{2n+1} = c_{n+1}^2 + c_n^2(q-1)/4.$$

The last formulas help in fast computing of c_n for large n. Substituting these values of a and b into the defining equation of c_n and using the binomial theorem we obtain

$$2^{n-1}c_n = \sum_k \binom{n}{2k+1} q^k.$$

Let n be a prime p, resp. $p+1$, where $p > 2$, and p differs from q, then we have

$$c_p \equiv q^{(p-1)/2} \mod p \quad \text{and} \quad 2c_{p+1} \equiv 1 + q^{(p-1)/2} \mod p.$$

Modulo p computing of c_p, c_{p+1} and $q^{(p-1)/2}$ makes primality testing possible, but pseudoprimes may occur.

2. $q \equiv 2$ or $3 \mod 4$

Now $P = 2$ and $Q = 1 - q$. Thus a and b are the roots of the equation

$$x^2 - 2x + 1 - q = 0,$$

namely

$$a = 1 + \sqrt{q} \qquad \text{and} \qquad b = 1 - \sqrt{q}.$$

The equation

$$c_{n+1} = 2c_n + c_{n-1}(q-1)$$

can easily be proved by induction. Similarly

$$c_{n+m} = c_{m+1}c_n + (c_{n+1} - 2c_n)c_m,$$

therefore

$$c_{2n} = 2c_n(c_{n+1} - c_n) \quad \text{and} \quad c_{2n+1} = c_{n+1}^2 + (q-1)c_n^2.$$

Substituting the values of a and b into the defining equation of c_n and using the binomial theorem we obtain

$$c_n = \sum_k \binom{n}{2k+1} q^k.$$

Let n be a prime p, resp. $p+1$, where p differs from q, we have

$$c_p \equiv q^{(p-1)/2} \mod p \quad \text{and} \quad c_{p+1} \equiv 1 + q^{(p-1)/2} \mod p.$$

Modulo p computing of c_p, c_{p+1} and $q^{(p-1)/2}$ makes the primality testing possible, but pseudoprimes may occur.

3. Programs for prime-testing

Either case has its own program. The first one runs for the primes $q \equiv \equiv 1 \mod 4$. With any prime $q < 50$ runs either of the programs.

We also want with our programs to decide if a candidate p is a prime or a pseudoprime.

Starting with 49 we test the numbers which are not divisible by a prime less than seven. We compute the values

$$c_p \mod p \qquad \text{and} \qquad c_{p+1} \mod p.$$

In the first program we compute $q^{(p-1)/2} \mod p$ before checking c_p and c_{p+1}.

If p is a prime $q^{(p-1)/2} \mod p$ should be equal to 1 or -1 (Fermat). This is the second filtering. Now comes a third filtering: whether p is a quadratic residue modulo q. We use here the reciprocity law of Gauss.

In the second program if $q = 2$, we use the values

$$2^{(p-1)/2} \equiv 1 \quad \text{if} \quad p \equiv 1 \mod 8 \quad \text{or} \quad p \equiv -1 \mod 8,$$
$$2^{(p-1)/2} \equiv -1 \quad \text{if} \quad p \equiv 3 \mod 8 \quad \text{or} \quad p \equiv -3 \mod 8.$$

If $q \equiv 3 \mod 4$, we again apply the recopricity law of Gauss. A candidate p passed the test will be checked by divisions if it is really prime. A prime d will be ranged among the divisors if

$$60k + 48 < d * d < 60k + 62, \qquad k = 0, 1, 2, 3, \ldots$$

as the square of a number which is not divisible by a prime less than seven has the form

$$120k + 49 \qquad \text{or} \qquad 120k + 1, \qquad k = 0, 1, 2, 3, \ldots$$

4. Results of program-runs

The programs run with the primes below 50 as parameters testing numbers less than 80 million. The programs were written in Euphoria programming language (version 2.4). The computer had an Intel Celeron CPU of 1.7 GHz. The running time was about twenty minutes per parameter.

The last prime was (the 4669382-nd) 79999987. (79999991=409∗195599 and 79999993=4229∗18917 and 79999999=1709∗46811)

The greatest prime below ten million was (the 664579-th) 9999991.

Here are the results of the first program ($q \equiv 1 \mod 4$).

The beginning of the sequence c_n is

$$1,\ 1,\ 2,\ 3,\ 5,\ 8,\ 13,\ 21,\ 34,\ 55,\ 89, \ldots \quad (q = 5).$$

This is the well-known Fibonacci sequence.

$$(q = 13) \quad 1,\ 1,\ 4,\ 7,\ 19,\ 40,\ 97,\ 217,\ 508, \ldots$$
$$(q = 17) \quad 1,\ 1,\ 5,\ 9,\ 29,\ 65,\ 181,\ 441, \ldots$$
$$(q = 29) \quad 1,\ 1,\ 8,\ 15,\ 71,\ 176,\ 673,\ 1905, \ldots$$
$$(q = 37) \quad 1,\ 1,\ 10,\ 19,\ 109,\ 280,\ 1261, \ldots$$
$$(q = 41) \quad 1,\ 1,\ 11,\ 21,\ 131,\ 341,\ 1651, \ldots$$

The next table shows the parameter q and the number of pseudoprimes below 10 million, rep. 80 million.

q	$< 10 \cdot 10^6$	$< 80 \cdot 10^6$
5	32	96
13	11	30
17	9	23
29	1	6
37	6	27
41	9	29

The pseudoprimes less than one million with $q = 5$ are

$$146611, \quad 252601, \quad 399001, \quad 512461, \quad 556421, \quad 852841.$$

The pseudoprime less than one million with $q = 13$ is

$$226801.$$

The pseudoprimes less than one million with $q = 17$ are

$$31621, \quad 188191, \quad 334153, \quad 683761.$$

The pseudoprime less than ten million with $q = 29$ is

$$1615681.$$

The pseudoprimes less than one million with $q = 37$ are

$$226801, \quad 410041, \quad 534061, \quad 765703, \quad 954271.$$

The pseudoprime less than one million with $q = 41$ is

$$512561.$$

The results of the second program ($q = 2$ or $q \equiv 3 \bmod 4$): the beginning of the sequences c_n are

$$
\begin{aligned}
(q = 2) \quad & 1,\ 2,\ 5,\ 12,\ 29,\ 70,\ 169, \ldots \\
(q = 3) \quad & 1,\ 2,\ 6,\ 16,\ 44,\ 120,\ 328, \ldots \\
(q = 7) \quad & 1,\ 2,\ 10,\ 32,\ 124,\ 440,\ 1624, \ldots \\
(q = 11) \quad & 1,\ 2,\ 14,\ 48,\ 236,\ 952,\ 4264, \ldots \\
(q = 19) \quad & 1,\ 2,\ 22,\ 80,\ 556,\ 2552,\ 15112, \ldots \\
(q = 23) \quad & 1,\ 2,\ 26,\ 96,\ 764,\ 3640,\ 24088, \ldots \\
(q = 31) \quad & 1,\ 2,\ 34,\ 128,\ 1276,\ 6392,\ 51064, \ldots \\
(q = 43) \quad & 1,\ 2,\ 46,\ 176,\ 2284,\ 11960, \ldots \\
(q = 47) \quad & 1,\ 2,\ 50,\ 192,\ 2684,\ 14200, \ldots
\end{aligned}
$$

The next table shows the parameter q and the number of pseudoprimes below 10 million, resp. 80 million.

q	$< 10 \cdot 10^6$	$< 80 \cdot 10^6$
2	316	846
3	20	48
7	5	13
11	11	22
19	9	20
23	14	25
31	11	22
43	9	22
47	18	33

The pseudoprimes less than one hundred thousand with $q = 2$ are

$$169,\ 961,\ 1121,\ 3827,\ 6601,\ 7801,$$
$$8119,\ 13067,\ 15841,\ 18241,\ 19097,\ 20833,$$
$$24727,\ 27971,\ 29953,\ 31417,\ 34561,\ 35459,$$
$$38081,\ 39059,\ 42127,\ 45961,\ 47321,\ 52633,$$
$$53041,\ 55969,\ 56953,\ 58241,\ 79361,\ 81361,$$
$$84587,\ 86033.$$

The pseudoprimes less than one million with $q = 3$ are

$$29341,\ 46657,\ 115921,\ 226801,\ 294409,$$
$$314821,\ 488881,\ 530881,\ 873181.$$

The pseudoprimes less than ten million with $q = 7$ are

$$1024651,\ 3581761,\ 4411681,\ 5444489,\ 8134561.$$

The pseudoprimes less than one million with $q = 11$ are

$$12403,\ 102173,\ 597871,\ 873181.$$

The pseudoprimes less than one million with $q = 19$ are

$$49141,\ 104653,\ 216457,\ 399001,\ 552721.$$

The pseudoprimes less than one million with $q = 23$ are

$$1729,\ 8911,\ 41041,\ 52633,\ 101101,$$
$$126217,\ 334153,\ 665281,\ 997633.$$

The pseudoprimes less than one million with $q = 31$ are

$$52801,\ 79003,\ 344641,\ 437251,\ 765703,\ 873181.$$

The pseudoprimes less than one million with $q = 43$ are

$$1729,\ 18721,\ 75361,\ 188461,\ 204001,\ 574561.$$

The pseudoprimes less than one million with $q = 47$ are

$$703,\ 1891,\ 399001,\ 552721.$$

Let us compare the number of pseudoprimes with the number of the Carmichael numbers.

Our number with $q = 2$ is greater than the number of the universal pseudoprimes (105 below ten million). The other parameters result less than the number of the Carmichael numbers. The best result we got are 1 and 5 with $q = 29$ and $q = 7$. There are pseudoprimes occuring with several parameters. See e.g. 1729 with $q_1 = 23$ and with $q_2 = 43$.

Here are all the mutual pseudoprimes per parameter-couples:

2	3			
		3363121	10402561	13694761
		23382529	46094401	50201089
		74927161		
2	5			
		5049001	5148001	7519441
		8719921	10024561	14609401
		27012001	62399041	68154001
2	7			
		10402561	17098369	
2	11			
		19384289	53245921	
2	13			
		6313681	10024561	75151441
2	17			
		26886817		
2	19			
		7207201		
2	23			
		52633	665281	10402561

2	29			
		1615681		
2	31			
		344641	4767841	23382529
		79411201		
2	37			
		410041	10024561	10402561
		26886817	68154001	
2	41			
		17098369		
2	43			
		4767841	7207201	
2	47			
		1615681	10024561	19384289
		68154001		
3	5			
		2704801	6189121	6840001
		10403641	14676481	15247621
		20964961	34657141	60957361
		62756641		
3	7			
		8134561	10402561	36765901
3	11			
		873181	53711113	
3	13			
		226801	15247621	20964961
		53711113	60957361	
3	17			
3	19			
		35703361		
3	23			
		10402561	20964961	
3	29			
3	31			
		873181	23382529	
3	37			
		226801	10402561	20964961
		26280073		
3	41			
		1461241	2433601	2704801
3	43			

3	47			
		2113921		
5	7			
		1024651	33596641	
5	11			
		4504501		
5	13			
		10024561	15247621	18307381
		20964961	43286881	60957361
5	17			
		3828001	13012651	
5	19			
		399001	5481451	17236801
		51803821		
5	23			
		1909001	20964961	
5	29			
		31405501		
5	31			
		3828001	9863461	33596641
		62289541		
5	37			
		10024561	16778881	20964961
		22187791	33796531	68154001
5	41			
		512461	1193221	2704801
5	43			
		1909001	16778881	17236801
		29111881	31405501	33596641
		34043101		
5	47			
		399001	1193221	2100901
		5481451	8341201	10024561
		10837321	17236801	27062101
		33302401	40430401	68154001
7	11			
		55462177		
7	13			
		3581761		
7	17			
7	19			

7	23			
		10402561		
7	29			
7	31			
		33596641		
7	37			
		10402561	19328653	55462177
7	41			
		17098369		
7	43			
		33596641		
7	47			
11	13			
		1152271	4335241	5968873
		6868261	53711113	54637831
11	17			
		67902031		
11	19			
11	23			
11	29			
11	31			
		873181	5968873	24550241
		67902031		
11	37			
		4335241	55462177	
11	41			
		3913003	67902031	
11	43			
		34901461		
11	47			
		1152271	4335241	5968873
		19384289		
13	17			
13	19			
13	23			
		20964961	40622401	
13	29			
13	31			
		5968873	14913991	40622401
13	37			
		226801	4335241	10024561
		17316001	20964961	

13	41			
		14913991	32914441	
13	43			
		11205601		
13	47			
		1152271	3057601	4335241
		5968873	10024561	11205601
		14913991		
17	19			
17	23			
		334153		
17	29			
17	31			
		3828001	43331401	67902031
17	37			
		26886817		
17	41			
		43331401	67902031	
17	43			
17	47			
		3375487		
19	23			
		14469841	75765313	
19	29			
19	31			
19	37			
		21459361		
19	41			
		14469841		
19	43			
		7207201	17236801	
19	47			
		399001	552721	5481451
		17236801		
23	29			
23	31			
		40622401		
23	37			
		10402561	20964961	
23	41			
		14469841		

23	43			
		1729	1909001	
23	47			
29	31			
29	37			
29	41			
		31470211		
29	43			
		31405501		
29	47			
		1615681		
31	37			
		765703		
31	41			
		14913991	43331401	67902031
31	43			
		4767841	33596641	
31	47			
		5968873	14913991	
37	41			
37	43			
		16778881		
37	47			
		4335241	10024561	68154001
41	43			
41	47			
		1193221	14913991	27402481
43	47			
		11205601	17236801	

As it can be seen, the next parameter couples have no mutual pseudoprime:

3, 17	3, 29	3, 43	7, 17
7, 19	7, 29	7, 47	11, 19
11, 23	11, 29	13, 17	13, 19
13, 29	17, 19	17, 29	17, 43
19, 29	19, 31	23, 29	23, 47
29, 31	29, 37	37, 41	41, 43

If there exists such a parameter-couple, which has no mutual pseudoprime below 10^{1000}, then our method is suitable for finding primes for public-key coding.

5. Additional program-runs

We got results for the q parameter values between 50 and 100, and between 100 and 200. The next tables show the number of pseudoprimes below 10 million, resp. 80 million. At the lefthand-side the parameter values (q) are of form $4k+1$.

q	$< 10 \cdot 10^6$	$< 80 \cdot 10^6$	q	$< 10 \cdot 10^6$	$< 80 \cdot 10^6$
53	10	25	59	9	25
61	10	15	67	11	23
73	3	14	71	14	23
89	4	19	79	14	22
97	7	28	83	9	24

Here are the results for the q parameter values between 100 and 200.

q	$< 10 \cdot 10^6$	$< 80 \cdot 10^6$	q	$< 10 \cdot 10^6$	$< 80 \cdot 10^6$
101	18	37	103	25	48
109	18	37	107	19	34
113	21	33	127	12	23
137	10	26	131	10	23
149	13	32	139	7	27
157	8	24	151	4	15
173	19	39	163	16	37
181	9	13	167	5	21
193	6	19	179	10	19
197	17	38	191	11	23
			199	3	15

6. Program-runs with other sequences

The $P = 2$ and $Q = 1 - q$ choice also works for q of form $4k + 1$. The sequences will differ from ones in point 1, of course. The number of pseudoprimes below 10 million, resp. 80 million:

q	$< 10 \cdot 10^6$	$< 80 \cdot 10^6$	q	$< 10 \cdot 10^6$	$< 80 \cdot 10^6$
5	29	66	53	10	27
13	13	34	61	11	15
17	9	25	73	3	15
29	3	9	89	6	20
37	9	31	97	7	26
41	11	29			

And here are the results for q parameter values between 100 and 200.

q	$< 10 \cdot 10^6$	$< 80 \cdot 10^6$	q	$< 10 \cdot 10^6$	$< 80 \cdot 10^6$
101	17	38	157	4	16
109	14	30	173	15	31
113	17	27	181	10	17
137	15	33	193	6	22
149	11	25	197	10	27

References

[1] **Lucas E.,** Théorie des fonctions numériques simplement périodiques, *Amer. J. Math.,* **1** (1) (1878), 184-240, 289-321.

[2] **Ribenboim P.,** *The new book of prime number records,* Springer Verlag, New York, 1979.

(Received October 16, 2008)

A. Krem
Budapest, Hungary
alajos.krem@freemail.hu