Information Sciences Letters
An International Journal

http://dx.doi.org/10.18576/isl/110430

A Novel Algorithmic Approach using Little Theorem of Fermat For Generating Primes and Poulet Numbers in Order

Abd Elhakeem Abd Elnaby[1] *and A. H. El-Baz* [2,*]

[1]Department of Mathematics, Faculty of Science, Damietta University, New Damietta, Egypt
[2]Department of Computer Science, Faculty of Computers and Artificial Intelligence, Damietta University, New Damietta, Egypt

Received: 2 Feb. 2022, Revised: 22 Mar. 2022, Accepted: 26 May 2022
Published online:1 Jul. 2022.

Abstract: Computer encryption are based mostly on primes, which are also vital for communications. The aim of this paper is to present a new explicit strategy for creating all primes and Poulet numbers in order up to a certain number by using the Fermat's little theorem. For this purpose, we construct a set C of odd composite numbers and transform Fermat's little theorem from primality test of a number to a generating set Q of odd primes and Poulet numbers. The set Q is sieved to separate the odd primes and the Poulet numbers. By this method, we can obtain all primes and Poulet numbers in order up to a certain number. Also, we obtain a closed form expression which precisely gives the number of primes up to a specific number. The pseudo-code of the proposed method is presented.

Keywords: Primes, Poulet numbers, Cryptography, Algorithms.

1 Introduction

In today's society, with the rapid progress of computers, number theory will make more prominent long strides in the future. As a principal direction, number theory spreads extraordinary impacts on other disciplines, and it is the basis of many directions. Primes are viewed as one of the primary intriguing subjects with regards to number theory and they have various applications in different disciplines e.g., cryptography [1].

Two integers c and d are congruent modulo n, written $c \equiv d \pmod{n}$ if n divides c-d, or equivalently, if $d = c + \text{some multiple of } n$. In a congruence $\bmod n$, the number n is called the modulus, $n \geq 2$ [2].

Little theorem of Fermat states that if p is a prime and d is any integer not divisible by p, then $d^{p-1} - 1$ is divisible by p [2]. A Poulet number is a composite odd number n such that $2^{n-1} \equiv 1 \pmod{n}$. A composite number $n \in N$ satisfying the congruence $2^{n-1} \equiv 1 \pmod{n}$ is called pseudoprime number (to base 2). Also, Poulet numbers are specific kinds of Fermat pseudoprime numbers themselves, namely to the base 2. It is clear from little theorem of Fermat and definition of a Poulet number that if q_i is an odd integer greater than one and $2^{q_i-1} - 1$ is divisible by q_i, then q_i is either an odd prime or a Poulet number.

In this article, we use Fermat's little theorem and definition of the Poulet number in order to construct a set Q which contains the odd primes and the Poulet numbers up to a certain number q_m, where $q_m \in Q$ represents the largest number in the set Q. Generating primes by using sieving methods is a fundamental topic in number theory. Also, we use the intersection and difference operations defined on the sets to sieve the primes and the Poulet numbers. Furthermore, we determine an explicit closed form mathematical expression which exactly gives the number of primes up to a certain number $q_m \in Q$. For the best of our knowledge, that is the second exact formula given in literature where the first one was given in [3].

Algorithms sieves of Eratosthenes and Sundaram are algorithms that used to delete composite numbers from a set of numbers that exist, these processes are quite good as algorithms that could be applied to different processes of cryptographic algorithms [4, 5]. Jo and Park [6] used GCD primality test to generate primes for mobile smart devices. Konigsberg [7] generated primes and twin primes using the divisibility properties of binomial expressions. Little

*Corresponding author e-mail: elbaz@du.edu.eg

theorem of Fermat is used for primality test of a number in Tarafder and Chakroborty [8]. Sah et al [9] studied the different proofs for little theorem of Fermat and applications implicated through the 17th -21th centuries.

The article is coordinated as follows: the suggested new technique for creating the primes and the Poulet numbers in order up to a certain number introduces in section 2. The Pseudocode for the suggested method is presented in section 3. Section 4 concludes the study.

2 A New Technique for Creating the Primes and the Poulet Numbers

Fermat's little theorem is used to generate the primes and the Poulet numbers in order up to a certain number.
The suggested method is composed of the following steps:
The first step: Let Q be the set of the odd primes and the Poulet numbers up to $q_m \in Q$, where q_m is the largest number generated in Q,

$$Q = \left\{ q_j : \frac{2^{q_j - 1} - 1}{q_j} \text{ is a positive integer, j} = 1, 2, 3, \ldots, \text{m} \right\}.$$

The second step: Create the set C_i, where

$$C_i = \left\{ (2i+1)(2i+2n_i+1) : n_i = 0,1,2,\ldots, \left\lfloor \frac{q_m - (2i+1)^2}{2(2i+1)} \right\rfloor \right\}$$
$$, i = 1,2,3,\ldots,k.$$

And $|C_i| = \left\lfloor \frac{q_m - (2i+1)^2}{2(2i+1)} \right\rfloor + 1$, where $|.|$ is the Cardinality and $\lfloor . \rfloor$ is the floor function.

If $i = k+1$, we obtain $\left\lfloor \frac{q_m - (2i+1)^2}{2(2i+1)} \right\rfloor < 0$ then stop generating the set C_i.

The third step: Let $C = \bigcup_{i=1}^{k} C_i$ and $T^{(q_m)} = Q \cap C$, where $T^{(q_m)}$ is the set of Poulet numbers up to q_m.

The fourth step: Let P be the set of primes up to $q_m \in Q$,

then $P = \{2\} \cup \left(Q - T^{(q_m)} \right)$.

Moreover, we can determine the number of primes up to a certain number q_m by using the following formula

$$|P| = 1 + |Q| - \left| T^{(q_m)} \right|.$$

The previous method can be transformed to the next theorem.

Theorem

Consider the set

$$Q = \left\{ q_j : \frac{2^{q_j - 1} - 1}{q_j} \text{ is a positive integer, j} = 1, 2, 3, \ldots, \text{m} \right\}.$$ Let P, $P^{(q_m)}$ and $T^{(q_m)}$ be sets of the primes, the odd primes and the Poulet numbers, respectively up to a certain number q_m, where q_m is the largest number generated in Q. If $C = \bigcup_{i=1}^{k} C_i$,

$$C_i = \left\{ (2i+1)(2i+2n_i+1) : n_i = 0,1,2,\ldots, \left\lfloor \frac{q_m - (2i+1)^2}{2(2i+1)} \right\rfloor \right\}$$
$$, i = 1,2,3,\ldots,k.$$

Then $P = \{2\} \cup P^{(q_m)} = \{2\} \cup \left(Q - T^{(q_m)} \right)$ and $|P| = 1 + |Q| - \left| T^{(q_m)} \right|$ where $|.|$ denotes the cardinality and $\lfloor . \rfloor$ is the floor function.

Proof:

According to Fermat's little theorem and definition of a Poulet number, then the set Q as stated can be written as follows

$$Q = P^{(q_m)} \bigcup T^{(q_m)} \quad (1.1)$$

Assume that C_i and C are as stated and by using [3], C can be written as follows

$$C = \{ 9 \le c \le q_m : c \text{ is an odd composite number} \}. \quad (1.2)$$

From definition of Q, a Poulet number and C, it is clear that

$$T^{(q_m)} = Q \cap C. \quad (1.3)$$

Also, from definition of a Poulet number and a prime, it is clear that $P^{(q_m)} \bigcap T^{(q_m)} = \varphi$. (1.4)

This implies that equation (1.1) gives

$$P^{(q_m)} = Q - T^{(q_m)}. \quad (1.5)$$

Since P and $P^{(q_m)}$ are sets of the primes and the odd primes, respectively, then

$$P = \{2\} \cup P^{(q_m)} = \{2\} \cup \left(Q - T^{(q_m)}\right). \quad (1.6)$$

Therefore, the first demand is proved.
Since $P^{(q_m)} \cap \{2\} = \varphi.$, then (1.6) gives

$$|P| = |\{2\}| + \left|P^{(q_m)}\right| = 1 + \left|Q - T^{(q_m)}\right| \quad (1.7)$$

Since $Q = \left(Q - T^{(q_m)}\right) \cup T^{(q_m)}$ and $\left(Q - T^{(q_m)}\right) \cap T^{(q_m)} = \varphi$, then

$$|Q| = \left|Q - T^{(q_m)}\right| + \left|T^{(q_m)}\right| \Rightarrow \left|Q - T^{(q_m)}\right| = |Q| - \left|T^{(q_m)}\right|. \quad (1.8)$$

By substituting from (1.8) in (1.7), we obtain

$$|P| = 1 + |Q| - \left|T^{(q_m)}\right|. \quad (1.9)$$

Therefore, the second demand is proved and hence the proof is completed.

3 Pseudocode for the Suggested Method

The next algorithm gives our suggested strategy for producing primes and Poulet numbers in order.

Algorithm: The suggested strategy for producing primes and Poulet numbers in order

```
1: function Prime_Poulet(m)
2:     for n from 3 by 2 to m do
3:         q ← (2^(n−1) − 1)/n
4:         if q is integer then
5:             i ← i+1
6:             Q[i] ← n
7:         end if
8:     end do
9: qm ← max(Q)                    ▷ qm is the limit up to which primes and Poulet
                                    numbers are generated
10:  k ← 1
11:  i ← 0
12:  while k > 0 do
13:      i ← i+1
14:      k ← ⌊(qm − (2i+1)^2)/(2(2i+1))⌋
15:  end do
16:  k ← i
17:  for i from 1 to k do
18:      d ← ⌊(qm − (2i+1)^2)/(2(2i+1))⌋
19:      for n from 0 to d do
20:          x[n+1] ← (2i+1)(2i+2n+1)
21           end do
22:      C ← x                    ▷ save the generated elements x in vector C
23:      x ← [ ]
24:  end do
25:  Poulet ← Q ∩ C               ▷ Poulet is the set of Poulet numbers
26:  p ← Q − Poulet               ▷ P is the set of primes
27: end function
```

4 Conclusions

An original precise procedure for creating the primes and poulet numbers is proposed which has numerous applications in many disciplines especially cryptography. This strategy relies upon the Fermat's little theorem. It tends to be utilized to produce all primes and poulet numbers up to a specific number. Additionally, a closed form expression which accurately decides the quantity of primes up to a specific number is given.

Acknowledgment

This work was supported by the Academy of Scientific Research and Technology (ASRT), Egypt, under Grant 6499.

Competing interests: The authors declare that they have no competing interests.

References

[1] D. R. Stinson. *Cryptography: Theory and Practice*. 4th Edition, CRC Press, (2019).

[2] A. Granville. *Number theory revealed: A master class*. American Mathematical Society, (2020).

[3] A. E. H. Abd Elnaby and A. H. El-Baz, A new explicit algorithmic method for generating the prime numbers in order*, Egyptian informatics journal*., **22**, 101-104 (2021).

[4] M. E. O'Neill, The Genuine Sieve of Eratosthenes*, J. of Functional Programming*, **19**, 95-106 (2009).

[5] M. Lambert, Calculating the Sieve of Eratosthenes, *J. of Functional Programming*, **14**, 759-763 (2004).

[6] H. Jo and H. Park, *Fast prime generation algorithms using proposed GCD test on mobile smart devices*, IEEE International Conference on Big Data and Smart Computing (BigComp), 18-20 Jan., 374-377, Hong Kong, China, (2016).

[7] Z. R. Konigsberg, *Primes and Twin Primes Generator Algorithms*, IEEE Proceedings of the Multiconference on "Computational Engineering in Systems Applications", 4-6 Oct., 1-4, Beijing, China, (2006).

[8] A. K. Tarafder and T. Chakroborty, *A comparative analysis of general sieve-of Eratosthenes and Rabin-Miller approach for prime number generation*, IEEE international conference on electrical, computer and communication engineering (ECCE), 1- 4, Cox'sBazar, Bangladesh, (2019).

[9] R. P. Sah, U. N. Roy, A. K. Sah and S. K. Sourabh, Early proofs of Fermat's little theorem and applications, *IJMTT*, **64**, 2, 74-79 (2018).