Criptografie-Tema4

guzurazvan

May 2024

24. Care sunt factorii primi ai numărului n = 16759?

$\left[\sqrt{}(16759) = 129.\right]$
$t = 130 : t^2 - n = 16900 - 16759 = 141 \neq s^2$
$t = 131 : t^2 - n = 17161 - 16759 = 402 \neq s^2$
....Nu am găsit un t astfel încât $t^2 - n = s^2$.

n=16759
Fie $a \in [2, n-1]$ un număr aleatoriu. Dacă $a^{n-1} = 1(modn)$ atunci n are o probabilitate mare să fie prim. În caz contrar, numarul n este compus.
Pentru $a = 2$ avem: $a^{n-1}(modn) = 2^{16758}(mod16759) = 1(mod16759)$. ⇒ n = 16759 este cel mai probabil prim.

16759 % 3 = 1;
16759 % 5 = 4;
16759 % 7 = 1;
16759 % 11 = 6;
16759 % 13 = 2;
16759 % 17 = 14;
16759 % 19 = 1;
16759 % 23 = 15;
16759 % 29 = 26;
16759 % 31 = 19;
16759 % 37 = 35;
16759 % 41 = 31;
16759 % 43 = 32;
16759 % 47 = 27;
16759 % 53 = 11;
16759 % 59 = 3;
16759 % 61 = 45;
16759 % 67 = 9;
16759 % 71 = 3;
16759 % 73 = 42;
16759 % 79 = 11;
16759 % 83 = 76;

16759 % 89 = 27;
16759 % 97 = 75;
16759 % 101 = 94;
16759 % 103 = 73;
16759 % 107 = 67;
16759 % 109 = 82;
16759 % 113 = 35;
16759 % 127 = 122;
Nu am găsit niciun factor prim între 2 și 129 pentru numărul 16759, deci este prim.

2. Studiați algoritmul de factorizare rho al lui Pollard și aplicați-l pentru 10909.

Algoritmul de factorizare rho al lui Pollard este o metodă eficientă pentru factorizarea numerelor compuse. Acesta se bazează pe conceptul de funcții iteratoare și pe faptul că secvențele mod n vor intra într-un ciclu. Algoritmul încearcă să găsească un divizor al numărului prin calcularea valorilor iteratoare și identificarea ciclurilor.
Algoritmul Rho al lui Pollard:

- **Alegeți o funcție de iterare:** $f(x) = x^2 + c$ mod n, unde c este o constantă.
- **Inițializați:** Alegeți valori inițiale pentru $x_1 și ix_2$ (de obicei $x_1 = x_2 = 2$).
- **Iterați:** Calculați valorile următoare ale lui $x_1 și ix_2$ folosind funcția de iterare.
- **Găsiți divizorul:** Calculați $d = cmmdc(|x_1 - x_2|, n)$. Dacă d este un divizor netrivial (adică $1 < d < n$, atunci d este un factor al lui n).

Pentru n = 10909:
1. $f(x) = x^2 + 1$(mod n)
2. $x_1 = 2$, $x_2 = 2$
3. $x_3 = 5, x_4 = 26, x_5 = 677$
4.$cmmdc$:
$x_1 = f(x_1) = 2^2 + 1 \bmod 10909 = 5$
$x_2 = f(x_2) = 5^2 + 1 \bmod 10909 = 26$
$x_3 = f(x_3) = 26^2 + 1 \bmod 10909 = 677$
$x_4 = f(x_4) = 277^2 + 1 \bmod 10909 = 8635$

$10909 : d = cmmdc(|x_i - x_j|, 10909)$
i = 1, j = 3: $\Rightarrow |2 - 5| = 3 \Rightarrow cmmdc(3, 10909) = 1$
i = 1, j = 4: $\Rightarrow |2 - 26| = 24 \Rightarrow cmmdc(24, 10909) = 1$
i = 1, j = 5: $\Rightarrow |2 - 577| = 575 \Rightarrow cmmdc(575, 10909) = 1$
i = 2, j = 3: $\Rightarrow |2 - 5| = 3 \Rightarrow cmmdc(3, 10909) = 1$

i = 2, j = 4: $\Rightarrow |2 - 26| = 24 \Rightarrow cmmdc(24, 10909) = 1$
i = 2, j = 5: $\Rightarrow |2 - 577| = 575 \Rightarrow cmmdc(575, 10909) = 1$
i = 3, j = 4: $\Rightarrow |5 - 26| = 21 \Rightarrow cmmdc(21, 10909) = 1$
i = 3, j = 5: $\Rightarrow |5 - 577| = 572 \Rightarrow cmmdc(572, 10909) = 1$
i = 4, j = 5: $\Rightarrow |26 - 577| = 551 \Rightarrow cmmdc(551, 10909) = 1$
$\Rightarrow n = 10909$ este prim.

3. Implementați algoritmul de factorizare Fermat.

```cpp
#include "Header.h"

void FactorizareFermat(long long n) {
    long long a = ceil(sqrt(n));
    long long b2 = a * a - n;
    long long b = sqrt(b2);

    while (b * b != b2) {
        a += 1;
        b2 = a * a - n;
        b = sqrt(b2);
    }

    long long factor1 = a + b;
    long long factor2 = a - b;

    cout << "Factorii lui " << n << " sunt " << factor1 << " si "
        << factor2 << endl;
}

int main()
{
    FactorizareFermat(16751);
}
```