Вероятностные алгоритмы определения простоты числа

Gagik T. Papikyan

RUDN University, Moscow, Russian Federation

Определение простоты числа

Тест Ферма

Algorithm [edit]

The algorithm can be written as follows:

Inputs: *n*: a value to test for primality, *n*>3; *k*: a parameter that determines the number of times to test for primality

Output: *composite* if *n* is composite, otherwise *probably prime*

Repeat *k* times:

Pick *a* randomly in the range [2, *n* – 2]

If $a^{n-1} \not\equiv 1 \pmod{n}$, then return *composite*

If composite is never returned: return *probably prime*

The *a* values 1 and *n*-1 are not used as the equality holds for all *n* and all odd *n* respectively, hence testing them adds no value.

Пример реализации теста Ферма

```
function fermaTest(n){
    // if(n<2) return false
    // if(n in [2,3]) return true
    for(let i =0;i<200;i++){
        const a = Math.random() * (n-4) + 2
        const r = Math.pow(a, n-1) % n
        if(r === 1) return true
    }
    return false
}
```

Другие алгоритмы

- Алгоритм Рабина
- Алгоритм Соловэя