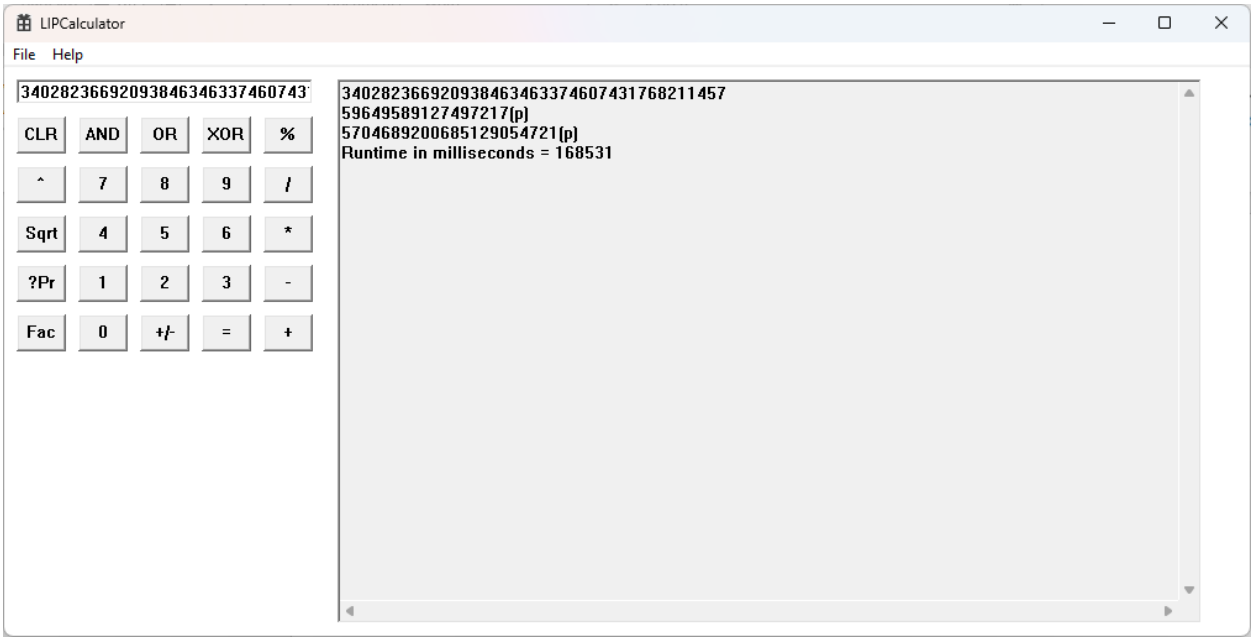
Blog Entry © Sunday October 19, 2025, by James Pate Williams, Jr. LIPCalculator (Large Integer Package Calculator)

LIP is a C-based large integer package developed by Arjen K. Lenstra while he was an employee of the Bell Telephone research laboratory in the 1980s. My LIPCalculator is a C/C++ large integer calculator using LIP. One of the first factoring tests was to factor the Seventh Fermat number (2^128+1):



[F_7] = 2^128 + 1
Factored into:
59649589127497217 (Prime, 56-bit)
570468920068512905472 1 (Prime, 72-bit)
Method: Pollard rho (≤ 10M iterations)
Runtime: 168,531 milliseconds

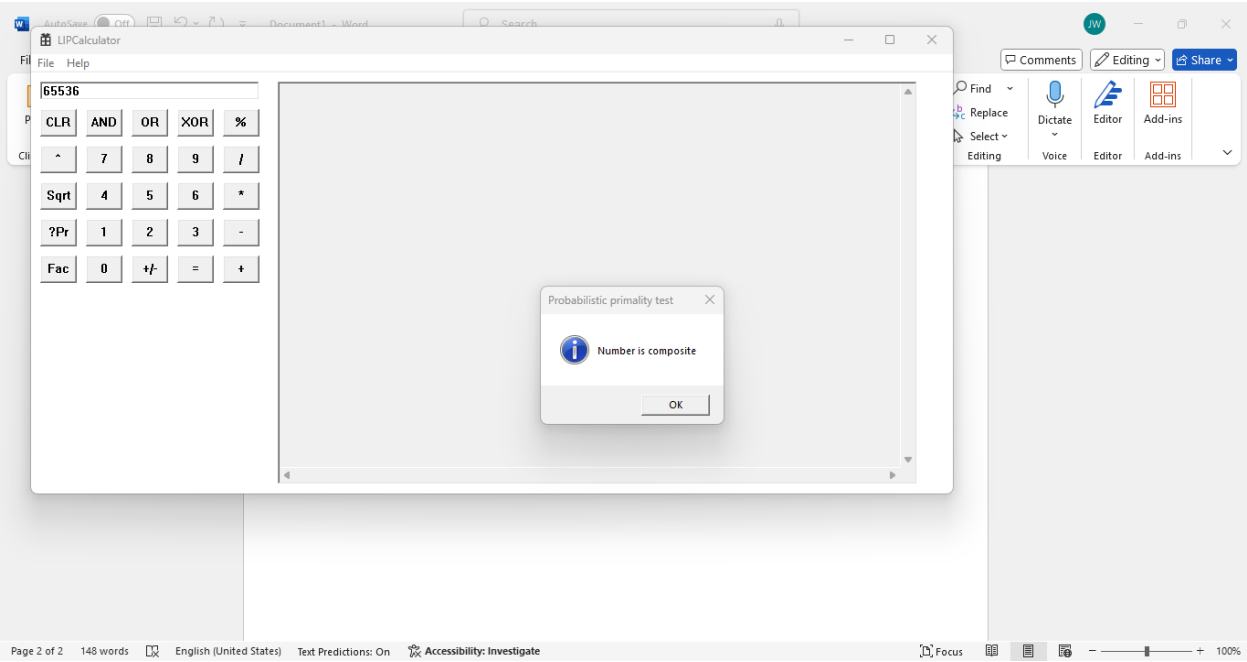
Pollard rho was used after trial division using a prime number bound of 10,000,000. The previous result was done using the Debug x64 Configuration. Next is done using the x64 Configuration

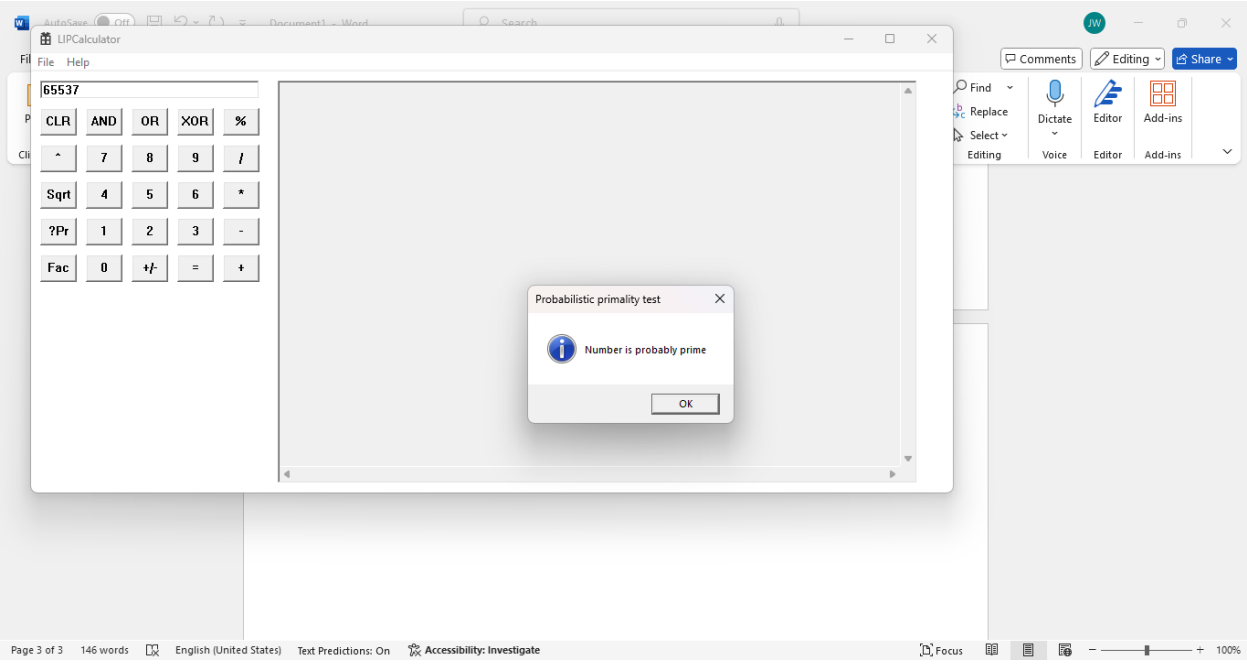
[F_7] = 2^128 + 1
→ Factored into:
59649589127497217 (Prime, 56-bit)
570468920068512905472l (Prime, 72-bit)
Method: Pollard rho (≤ 10M iterations)
Runtime: 54,092 milliseconds (Release x64)
Speedup: ~3× over Debug x64

The previous number was factored using trial division with a prime number bound of 10,000,000 and verified using the Microsoft calculator.

```
[12345678901234567890123456789]
Factored into 11 primes
Method: Trial Division (≤ 10M)
Runtime: 3 milliseconds Debug x64 Configuration
```

Next, we test whether a number is prime.





Now we use the square root calculator on 65536:

Now we compute 100001 * 1000000001:


Calculator
Scientific
100001 × 1000000001 =
100,001,000,100,001
DEG
F-E
MC
MR
M+
M−
MS
M
Trigonometry
Function
2nd
π
e
CE
x²
1/x
|x|
exp
mod
∛x
(
)
n!
÷
x^y
7
8
9
×
10^x
4
5
6
−
log
1
2
3
+
ln
+/−
0
.
=