

PrimeLister v2.9

Finally, it's here:

**PrimeLister** goes multi-core!

Let's dive straight into the benchmarks across three different machines:

CPU	Cores/Threads	Release
Intel Core i7-9750H	6 / 12	2019
Intel Core Ultra 7 155U	12 / 14	2023
Intel Xeon w7-3455	24 / 48	2023

If you're interested in how exactly PrimeLister generates prime numbers, check out the dedicated videos on our channel:

<https://www.youtube.com/@StrangerThings-jw3up>

“ImageJ, Prime Numbers and some C++ - A Prime Generator Tutorial”, Part 1 and 2

## Benchmark 1 (Slow), PrimeLister 2.6 Multicore:

Processor: Intel(R) Core(TM) i7-9750H CPU, 2.60GHz, 6 Cores, 12 Threads  
RAM: 16 GB  
OS: Win 11 (64 bit)

Calculate primes below	Primes found [N]	Last prime	Time [sec]	Primes/second
1,000,000 (1 M)	78,498 ✓*	999,983 ✓	0.006	13,554,937
100,000,000 (100 M)	5,761,455 ✓	99,999,989 ✓	0.143	40,281,022
1,000,000,000 (1 G)	50,847,534 ✓	999,999,937 ✓	0.927	54,824,722
2,000,000,000 (2 G)	98,222,287 ✓	1,999,999,973 ✓	2.270	43,274,875
5,000,000,000 (5 G)	234,954,223 ✓	4,999,999,937 ✓	12.086	19,440,458
10,000,000,000 (10 G)	455,052,511 ✓	9,999,999,967 ✓	38.078	11,950,599

\* <https://t5k.org/nthprime/>

## Benchmark 2 (Medium), PrimeLister 2.6 Multicore :

Processor: Intel(R) Core(TM) Ultra 7 155U, 1.70 GHz, 12 Cores, 14 Threads  
RAM: 64 GB  
OS: Win 11 (64 bit)

Calculate primes below	Primes found [N]	Last prime	Time [sec]	Primes/second
1,000,000 (1 M)	78,498 ✓*	999,983 ✓	0.006	13,558,215
100,000,000 (100 M)	5,761,455 ✓	99,999,989 ✓	0.125	46,271,172
1,000,000,000 (1 G)	50,847,534 ✓	999,999,937 ✓	0.588	86,509,280
2,000,000,000 (2 G)	98,222,287 ✓	1,999,999,973 ✓	1.343	73,112,848
5,000,000,000 (5 G)	234,954,223 ✓	4,999,999,937 ✓	4.105	57,240,641
10,000,000,000 (10 G)	455,052,511 ✓	9,999,999,967 ✓	12.850	35,411,309

\* <https://t5k.org/nthprime/>

## Benchmark 3 (Armageddon), PrimeLister 2.6 Multicore:

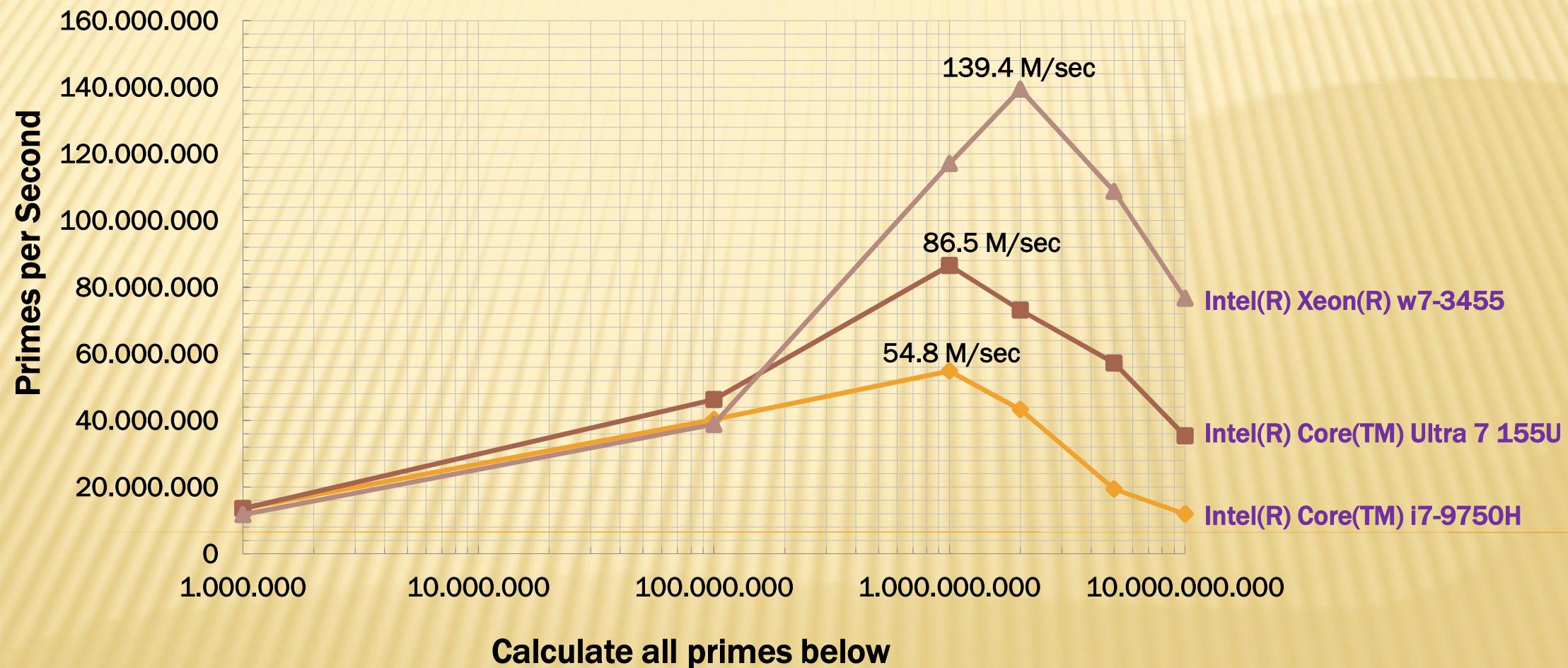
Processor: Intel(R) Xeon(R) w7-3455, 2.50 GHz, 24 Cores, 48 Threads  
RAM: 512 GB  
OS: Win 11 (64 bit)

Calculate primes below	Primes found [N]	Last prime	Time [sec]	Primes/second
1,000,000 (1 M)	78,498 ✓*	999,983 ✓	0.007	11,750,493
100,000,000 (100 M)	5,761,455 ✓	99,999,989 ✓	0.148	38,825,102
1,000,000,000 (1 G)	50,847,534 ✓	999,999,937 ✓	0.434	117,081,281
2,000,000,000 (2 G)	98,222,287 ✓	1,999,999,973 ✓	0.705	139,401,268
5,000,000,000 (5 G)	234,954,223 ✓	4,999,999,937 ✓	2.161	108,716,748
10,000,000,000 (10 G)	455,052,511 ✓	9,999,999,967 ✓	5.935	76,678,518

\* <https://t5k.org/nthprime/>

All 3 Benchmarks compared,  
Primelister 2.6 Multicore:

CPU	Cores/Threads	Release
Intel Core i7-9750H	6 / 12	2019
Intel Core Ultra 7 155U	12 / 14	2023
Intel Xeon w7-3455	24 / 48	2023



=> Create PrimeListerMulticore (**HardwareOptimization**) v.2.9:

**This code dynamically adapts to the hardware it is running on using**

**Automated Performance Calibration** – Benchmarks the CPU at startup to identify the optimal thread count and “sweet spot” for the specific architecture.

**Adaptive Load Balancing** – Dynamically scales processing segments based on the number of available logical cores and the total workload size.

**Cache-Centric Memory Management** – Constraints data chunks to 1–16 MB to ensure they fit within L2/L3 caches, minimizing high-latency RAM access.

**Native Hardware Intrinsics** – Leverages architecture-specific CPU instructions (like `_BitScanForward64`) for accelerated bit-level prime extraction.

## Benchmark 1 (Slow), PrimeLister 2.9 Multicore:

**Processor:** Intel(R) Core(TM) i7-9750H CPU, 2.60GHz, 6 Cores, 12 Threads  
**RAM:** 16 GB  
**OS:** Win 11 (64 bit)

Calculate primes below	Primes found [N]	Last prime	Time [sec]	Primes/second
1,000,000 (1 M)	78,498 ✓*	999,983 ✓	0.002225	35,279,999
100,000,000 (100 M)	5,761,455 ✓	99,999,989 ✓	0.075936	75,872,411
1,000,000,000 (1 G)	50,847,534 ✓	999,999,937 ✓	0.833296	61,019,767
2,000,000,000 (2 G)	98,222,287 ✓	1,999,999,973 ✓	2.100668	46,757,637
5,000,000,000 (5 G)	234,954,223 ✓	4,999,999,937 ✓	11.027764	21,305,698
10,000,000,000 (10 G)	455,052,511 ✓	9,999,999,967 ✓	37.950280	11,990,755

\* <https://t5k.org/nthprime/>

## Benchmark 2 (Medium), PrimeLister 2.9 Multicore:

Processor: Intel(R) Core(TM) Ultra 7 155U, 1.70 GHz, 12 Cores, 14 Threads  
RAM: 64 GB  
OS: Win 11 (64 bit)

Calculate primes below	Primes found [N]	Last prime	Time [sec]	Primes/second
1,000,000 (1 M)	78,498 ✓*	999,983 ✓	0.003470	22,620,598
100,000,000 (100 M)	5,761,455 ✓	99,999,989 ✓	0.050684	113,673,142
1,000,000,000 (1 G)	50,847,534 ✓	999,999,937 ✓	0.417282	121,854,039
2,000,000,000 (2 G)	98,222,287 ✓	1,999,999,973 ✓	0.948372	103,569,420
5,000,000,000 (5 G)	234,954,223 ✓	4,999,999,937 ✓	3.781640	62,130,239
10,000,000,000 (10 G)	455,052,511 ✓	9,999,999,967 ✓	12.643885	35,989,927

\* <https://t5k.org/nthprime/>

## Benchmark 3 (Armageddon), PrimeLister 2.9 Multicore:

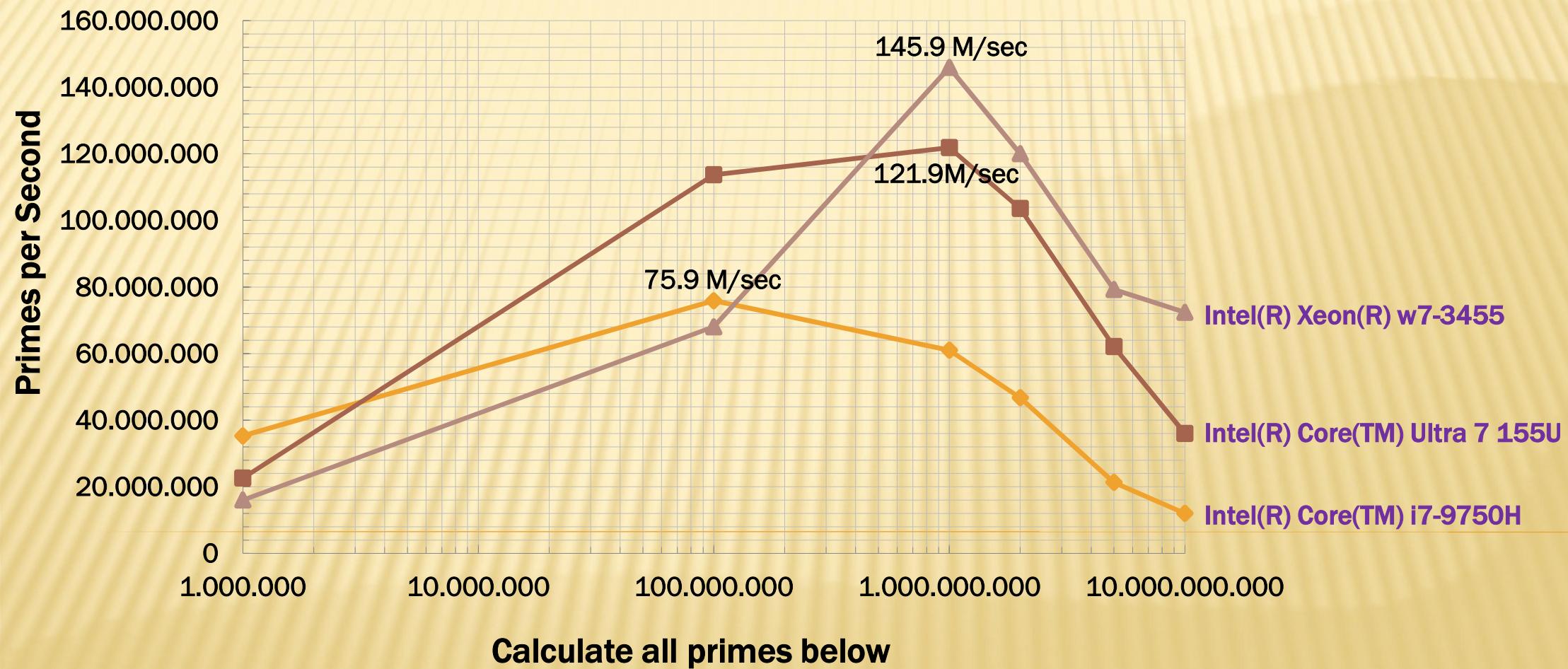
Processor: Intel(R) Xeon(R) w7-3455, 2.50 GHz, 24 Cores, 48 Threads  
RAM: 512 GB  
OS: Win 11 (64 bit)

Calculate primes below	Primes found [N]	Last prime	Time [sec]	Primes/second
1,000,000 (1 M)	78,498 ✓*	999,983 ✓	0.004901	16,016,731
100,000,000 (100 M)	5,761,455 ✓	99,999,989 ✓	0.084670	68,046,243
1,000,000,000 (1 G)	50,847,534 ✓	999,999,937 ✓	0.348499	145,904,389
2,000,000,000 (2 G)	98,222,287 ✓	1,999,999,973 ✓	0.818322	120,028,926
5,000,000,000 (5 G)	234,954,223 ✓	4,999,999,937 ✓	2.964506	79,255,786
10,000,000,000 (10 G)	455,052,511 ✓	9,999,999,967 ✓	6.289167	72,354,973

\* <https://t5k.org/nthprime/>

All 3 Benchmarks compared,  
PrimeLister 2.9 Multicore:

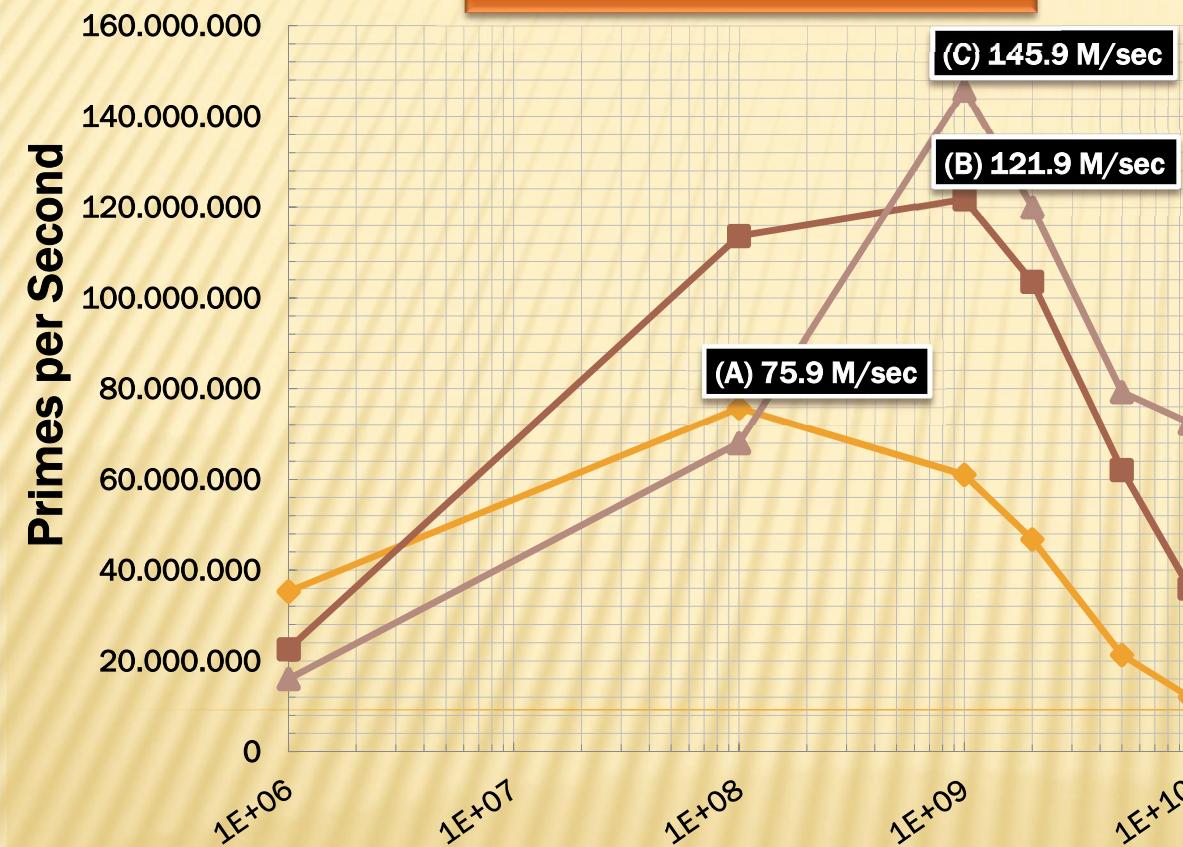
CPU	Cores/Threads	Release
Intel Core i7-9750H	6 / 12	2019
Intel Core Ultra 7 155U	12 / 14	2023
Intel Xeon w7-3455	24 / 48	2023



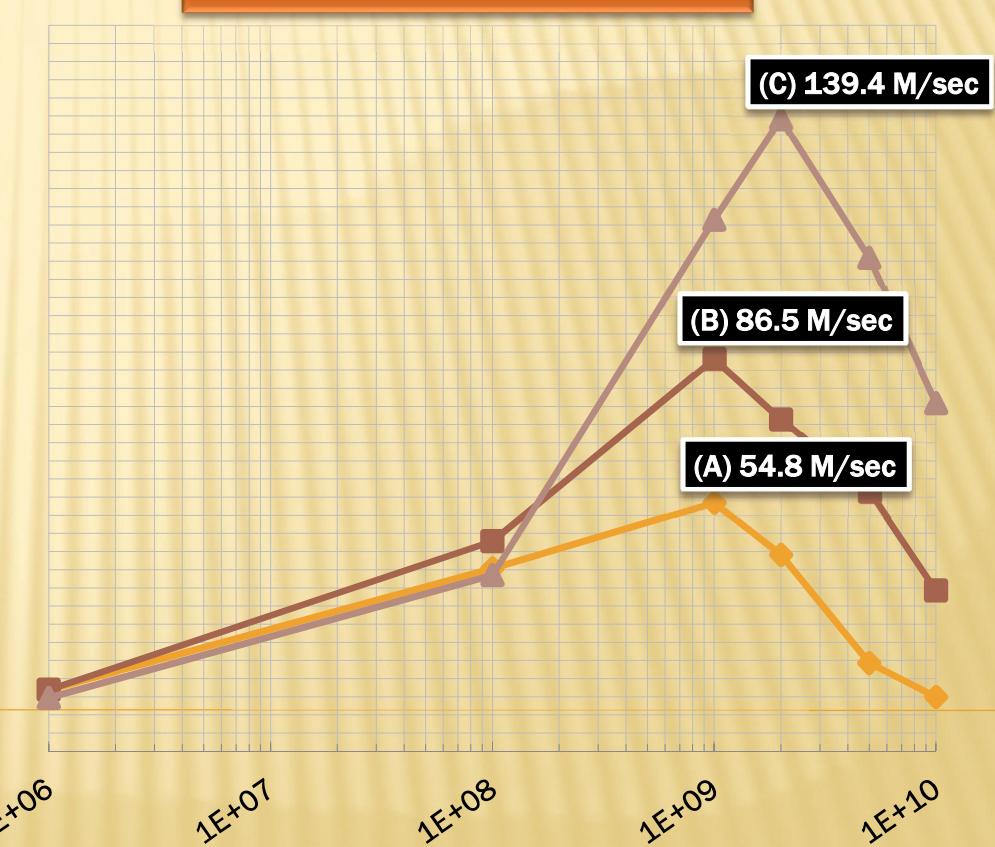
All 3 Benchmarks compared:

CPU	Cores/Threads	Release
(A) Intel Core i7-9750H	6 / 12	2019
(B) Intel Core Ultra 7 155U	12 / 14	2023
(C) Intel Xeon w7-3455	24 / 48	2023

PrimeLister 2.9 Multicore



PrimeLister 2.6 Multicore



Calculate all primes below

**Here you can find all the codes, compiled files (VisualStudio 2026) and additional materials:**

**<https://github.com/StrangestThings/PrimeLister-2.6-Multicore>**

<b>PrimeLister 2.6 multicore.cpp</b>	=> C++ source code
<b>PrimeLister SingleArray (ImageJ version).txt</b>	=> First version (ImageJ macro), better to understand
<b>PrimeListerMulticore v.2.6.exe</b>	=> Compiled multicore version 2.6
<b>README.md</b>	=> The story behind it

**<https://github.com/StrangestThings/PrimeLister-2.9-Multicore>**

<b>PrimeLister 2.9 multicore.cpp</b>	=> C++ source code
<b>PrimeLister SingleArray (ImageJ version).txt</b>	=> First version (ImageJ macro), better to understand
<b>PrimeListerMulticore v.2.9.exe</b>	=> Compiled multicore version 2.9
<b>README.md</b>	=> The story behind it