



APL+Win 19.0
Copyright (c) 2019 APLNow LLC.
All Rights Reserved
Apr 23, 2019

The APL+Win v19.0 release is available and recommended for all current APL+Win subscribers. To obtain this release, visit <http://www.apl2000.com/software.php> and click [APL+Win v19.0 Installer](#).

This release contains the enhancements and bug fixes listed below:

```
=====
** What's New/Fixed in 19.0.01 **
=====
```

New: The saved workspace version number reported by `⎕SYS[20]` is the integer value 55 in this release.

New: Support for EnableSSE2 is production ready in this release and easily controlled in the APLW.INI configuration file setting below (note the section name change to Config from Experimental):

```
[Config]
EnableSSE2=<value>
```

When <value> is set to 0 (still the default value), the previous behaviour of APL+Win will apply. When this setting is set to 1, when appropriate, the APL+Win interpreter will use the SSE2 instruction set to optimize array-based arithmetic operations, + - * ÷. Refer to the DEMO_SSE2.W3 workspace included in this release to test this enhancement.

Why is SSE2 disabled by default?

SSE2 floating point arithmetic operations, + - * ÷, can yield results that are very slightly different than traditional APL+Win result values. In most cases results will be identical. But in some cases very slight rounding differences can cause the low-order bits of the result value to be different. These possible result differences are limited to the last digit of precision in APL+Win and may not be apparent unless `⎕CT` is set sufficiently small or `⎕PP` is set sufficient high. However, you should be aware that result values may not be exactly the same with and without using the SSE2 optimization, even though you are unlikely to notice this difference with default `⎕CT` and `⎕PP` settings.



New: Two new high resolution timer system functions: `⌈TT` and `⌈TF`

`⌈TF`: Returns the frequency of the performance counter in terms of ticks.

`⌈TT`: Returns the number of ticks since the start of the current session.

Together, they aid in computing timing information for an APL function. An advantage of using the higher resolution `⌈TT` timer rather than `⌈AI` to capture timing information, `⌈TT` gives a better idea of what's actually happening on really fast execution runs. The `⌈AI` timer is only accurate to within about 1/1000 second while the `⌈TT` timer has a resolution of 3312526 ticks per second. That makes the accuracy of `⌈TT` run more accurate by a factor of about 3312 times more than `⌈AI[2]`.

Dividing `⌈TT` by `⌈TF`, the result will be the elapsed time in seconds since the APL+Win session started; basically the same value as `⌈AI[2]` but without the added overhead of computing the other elements of `⌈AI` and without the need to extract the second element of `⌈AI` to access the time information.

New: The Enclose operation was updated to avoid a potential crash arising from the following two conditions:

1. A workspace corruption during a workspace compaction
2. A memory leak during a report of a WS Full (workspace full) condition

Bug Fix: With support for EnableSSE2 enabled, floating point division operation could cause the session to suspend (hang) if the divisor is a vector of two or more elements, the vector contains at least one 0 and any one of the 0s is preceded by a non-zero element.

Example,

$$(0.2 \ 0.3) \div (0.1 \ 0)$$

Bug Fix: A `⌈LOCK` of an associated function caused abnormal termination in APL+Win 10 and newer versions.

Please send your comments or questions on this release to support@apl2000.com.

APL2000 Staff