Instead of “null”:

* Uninitialized (i.e. initialized to null)
* Explicitly Set To “No Value”
* A value that signifies that the Object has been deleted

The main issue is that a destroyed object and a null value are two different things. What is the definition of “destroyed”?

Concerns with “free-ing” an object:

* Could present a security concern (passing in an object that one does not want the callee to free)

Or to allow a null value:

String?[] array = …

Local variable declaration:

ENTER reg-id, reg-count

DECL CONST\_CLASS, CONST\_\*, … (debug info)

EXIT

-> is CONST\_\* defaultable (zero etc. for specific types)

constants

for each of the signed- and unsigned-integer types, 16-, 32-, 64- and 128-bit:

* min (signed integers only)
* -1 (signed integers only)
* 0
* 1
* 2
* max

**decimal** floating point:

binary **float**ing point:

for boolean:

* false
* true

for null:

* null