

NULL

Nullable < Thing > x

x = something

x = null

if (x is null) {
 // x is null

} else {

 // x is a thing

}

0 null false

Uninitialized memory

array values
/ object fields

1. clear some memory

2. compiler error for uninitialized var
proof

Correct \supset Provably correct

buffer overflow (out-of-bounds memory)

run-time

compile-time

Check bounds every time ~~~~~ fence check
 / \
 and error and nullx

no arrays exist

if ($i < 0$ || $i \geq a.length$) throw ArrayIndexOutOfBoundsException (),
 a[i]

for ($i = 0; i < n, i++$) ... a[i] ...

malloc / free

```
foo() {  
    :  
    :  
    int *x = malloc( ... )  
    :  
    int y = sum(x)  
    :  
    free(x)  
}
```