

Beyond the Basics

Amit Bijlani



Primitive Data Types

- char
- int
- long
- float
- double

Data Type Ranges

Data Type	Bytes	Range	Format
char	1	-128 to 127	%c
int	2	-32768 to 32767	%d
unsigned int	2	0 to 65535	%u
long	4	-2147483648 to 2147483647	%ld
float	4	3.4E -38 to 3.4E +38	%f
double	8	1.7E -308 to 1.7E +308	%lf

Please note these numbers are for a 16 Bit system

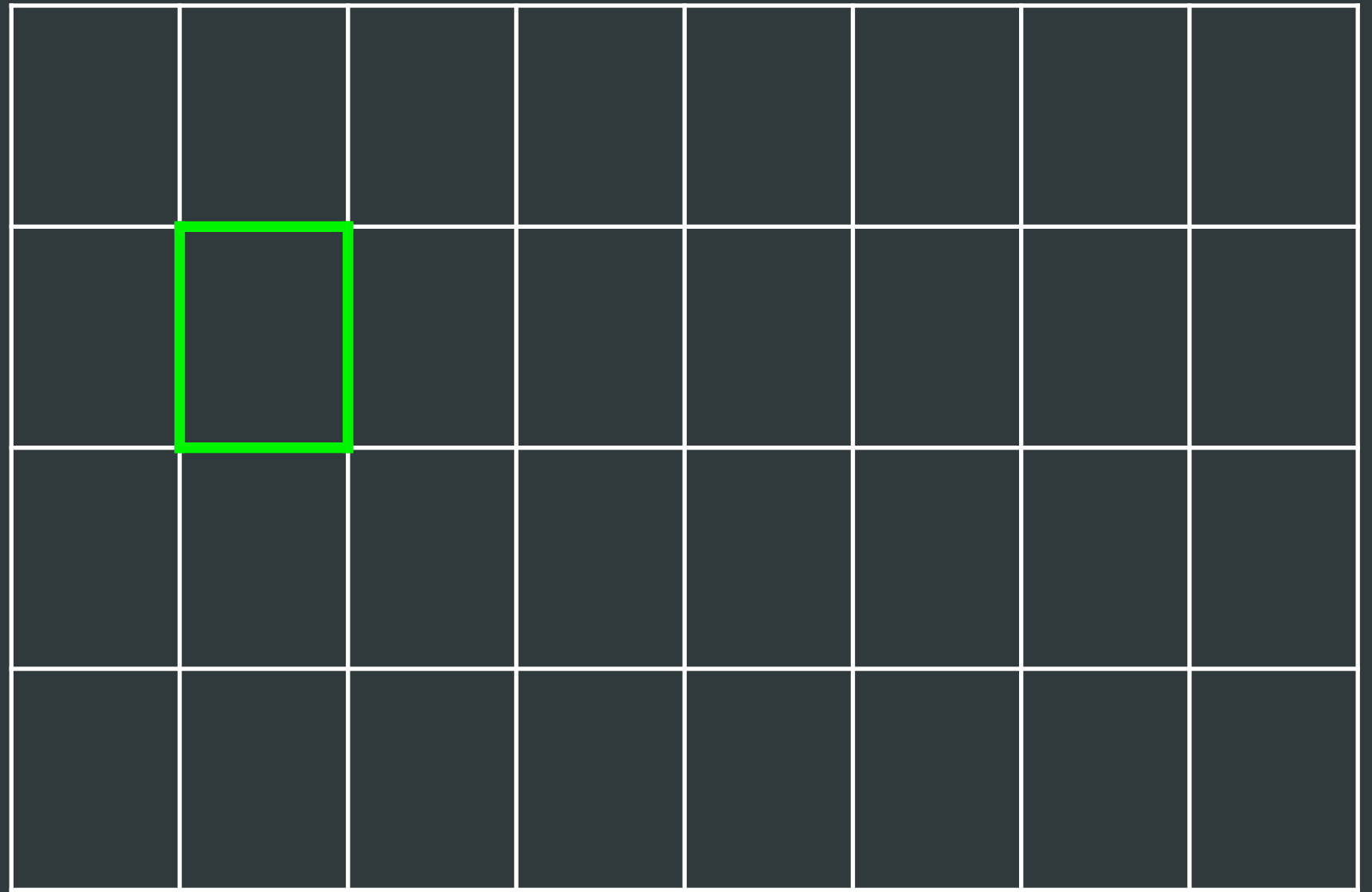
Primitive Data Types

- Fixed Memory
- Automatic Storage

Storing Primitive Types

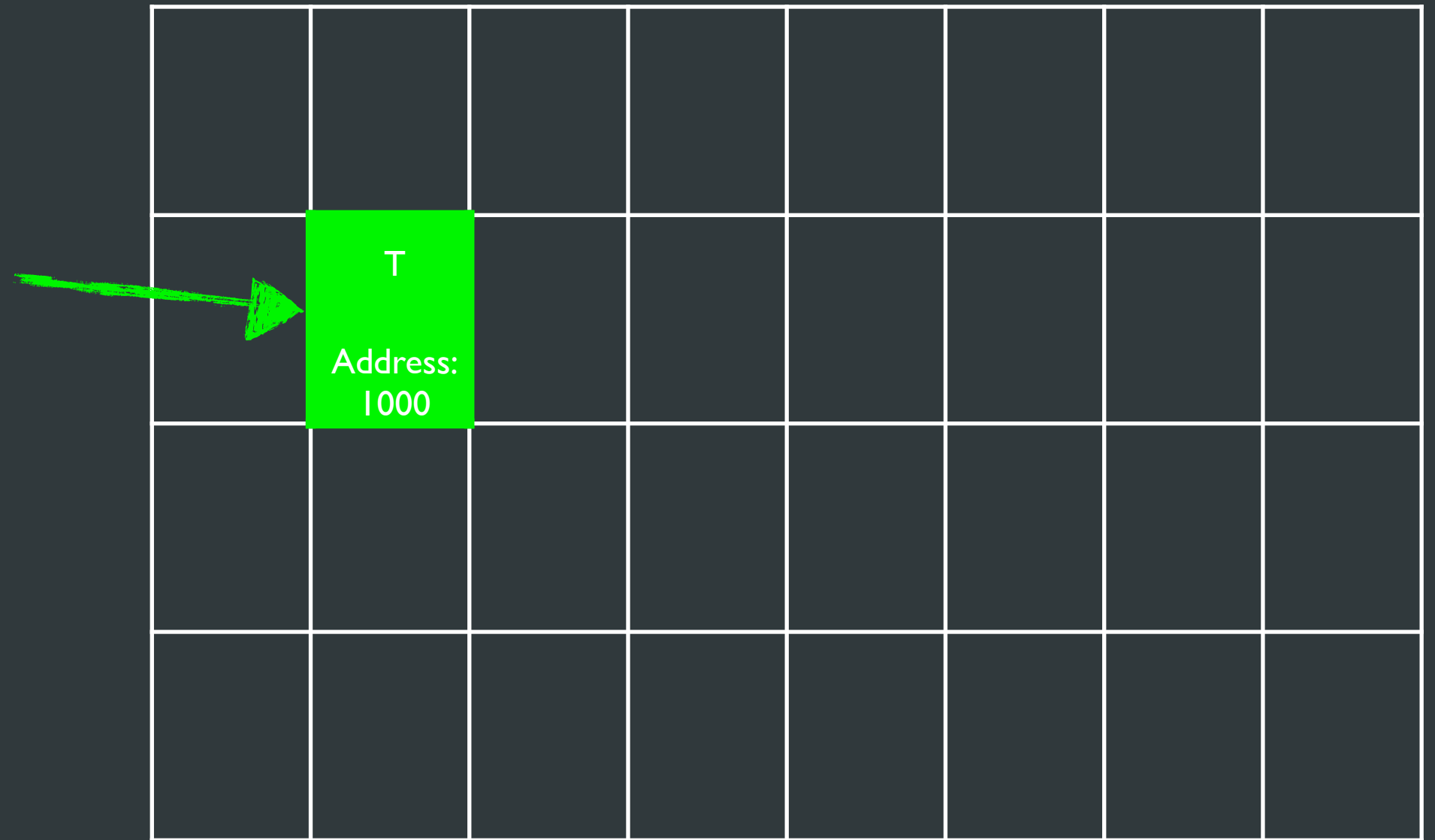
Storing Primitive Types

```
char c;
```



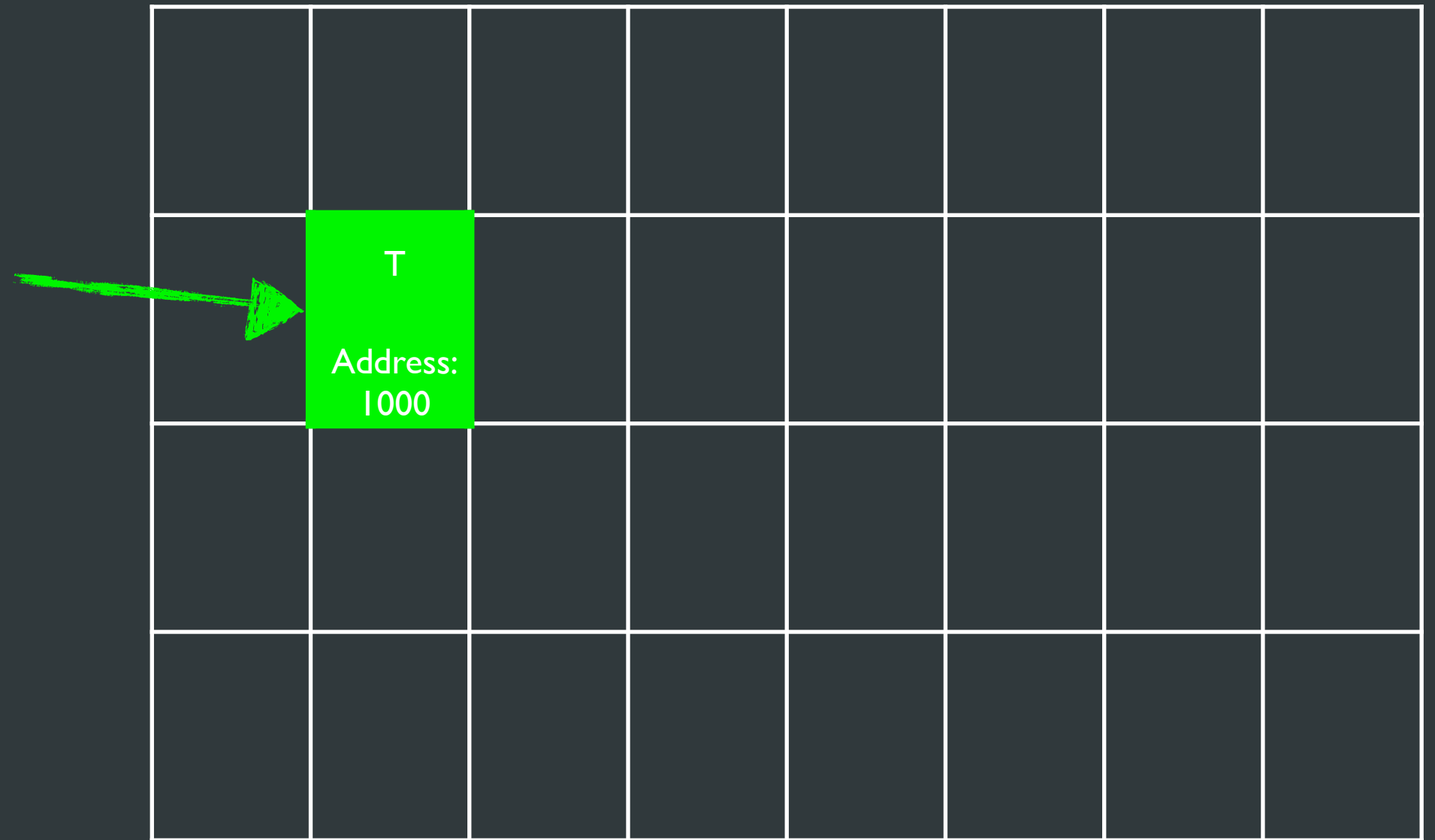
Storing Primitive Types

```
char c;  
c = 'T';
```



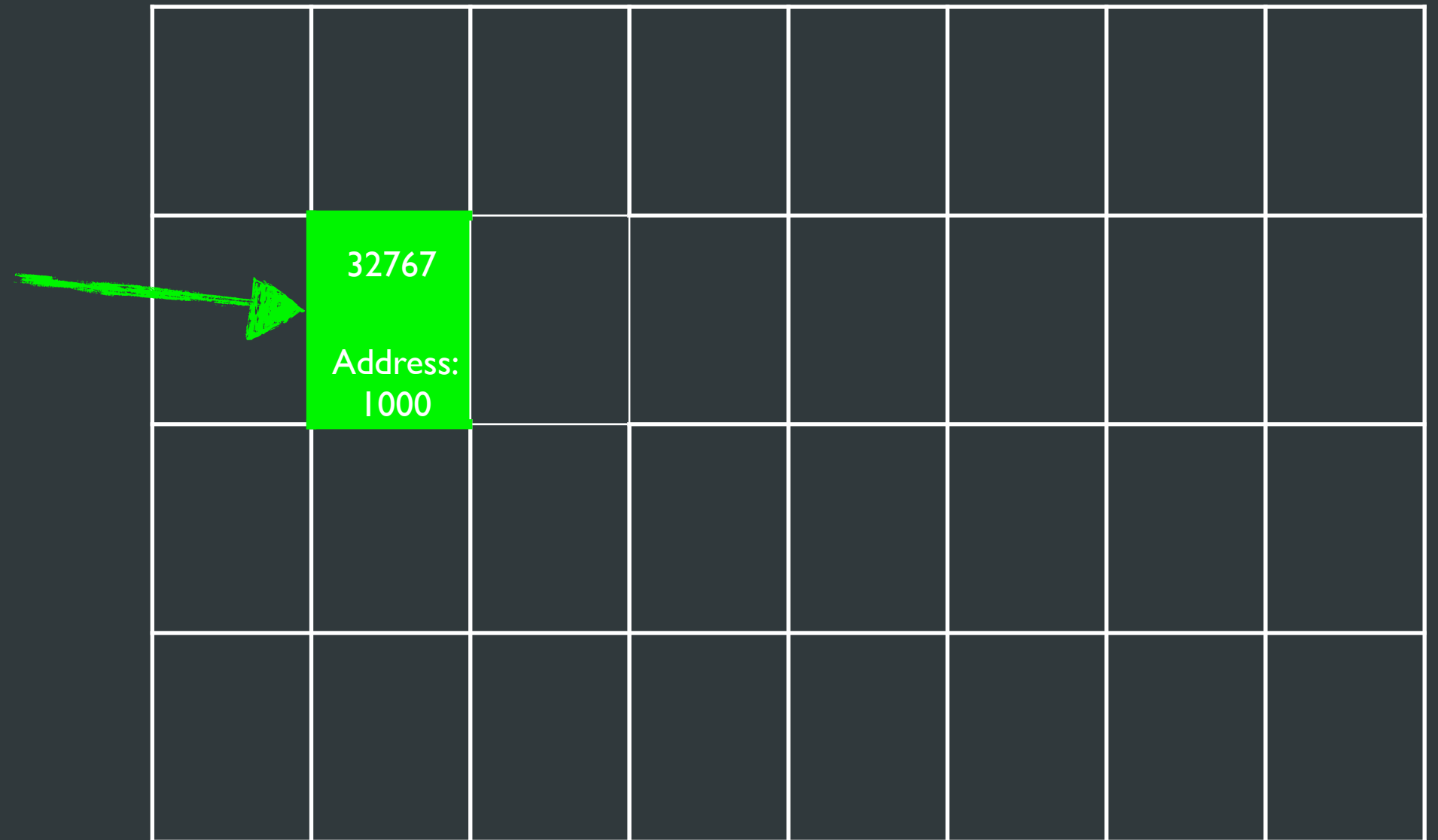
Storing Primitive Types

```
char c = 'T';
```



Storing Primitive Types

```
int max = 32767;
```



Pointers *

Pointers

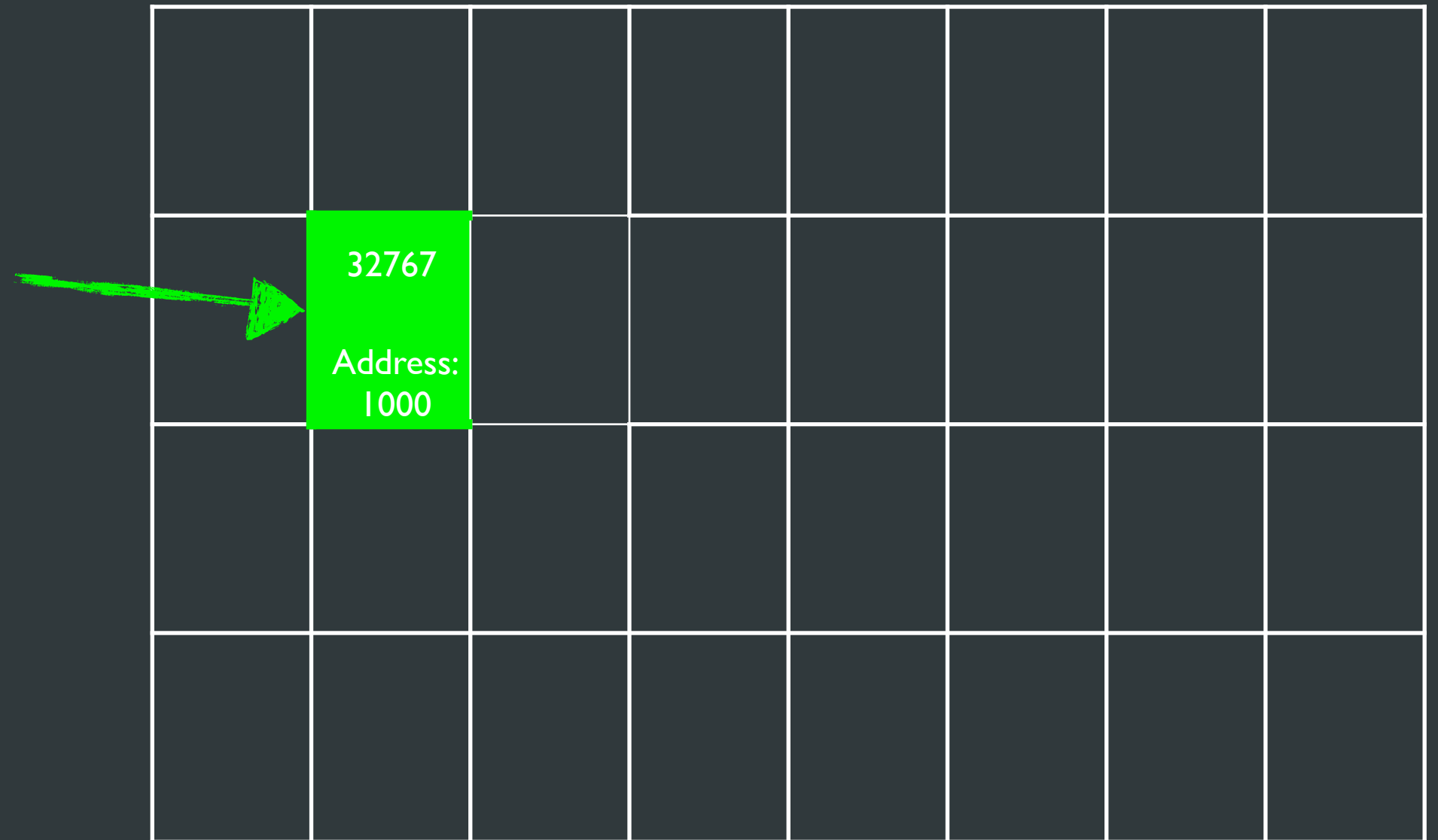
- Dynamic Memory
- Variable Storage

Pointers

```
int max = 32767;
```

```
// memory address
```

```
// &max
```



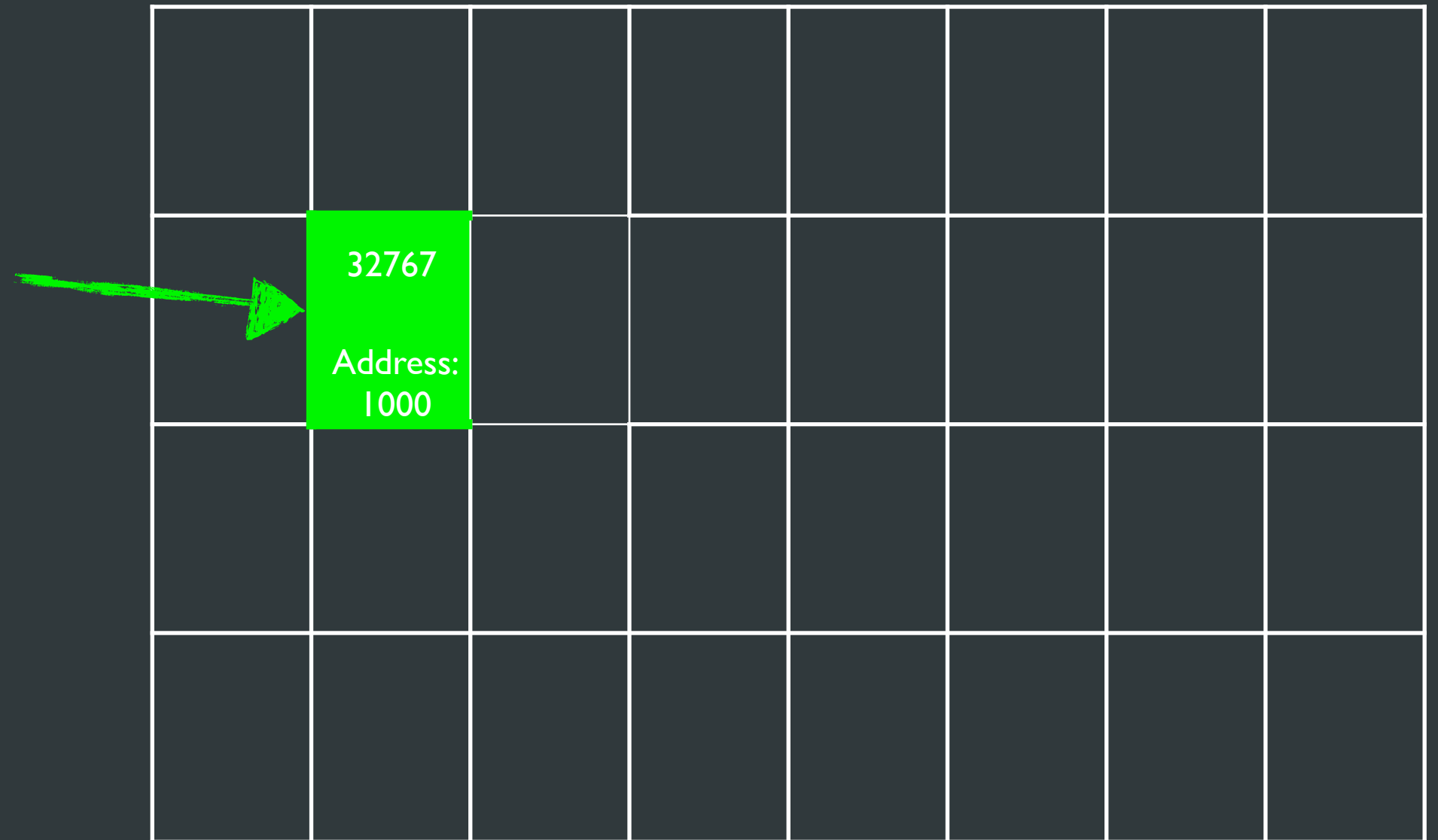
Pointers

```
int max = 32767;
```

```
// memory address  
// &max
```

```
int *p;  
p = &max
```

```
// p == 1000  
// *p == 32767
```



Why Pointers?

Pointers

```
int i = 100;
```

```
increment(int x);
```



Pointers

```
int i = 100;
```

```
increment(int *x);
```

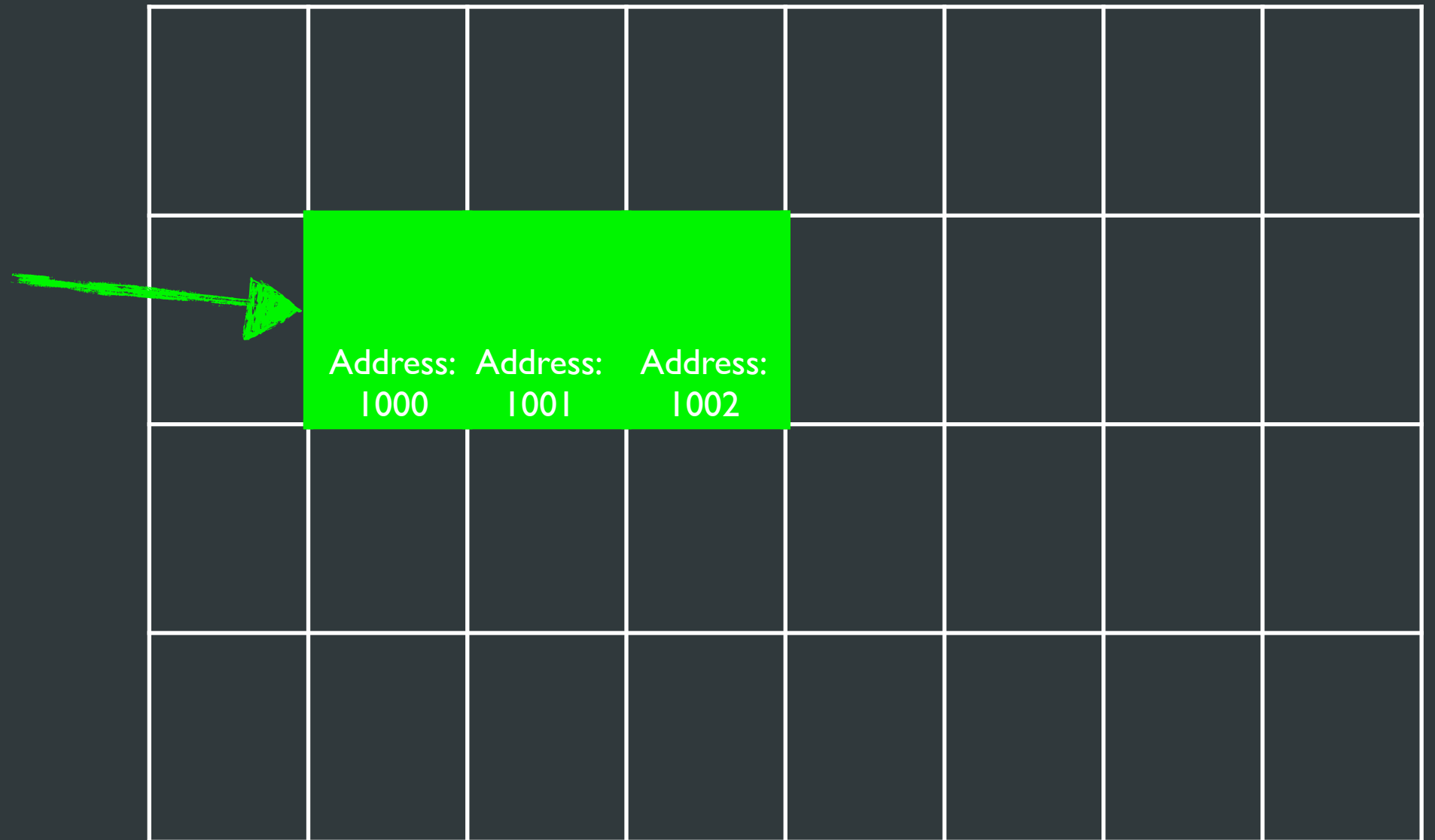


Arrays

Arrays

```
int primes[3];
```

```
// primes[0]  
// memory address  
// +  
// offset
```



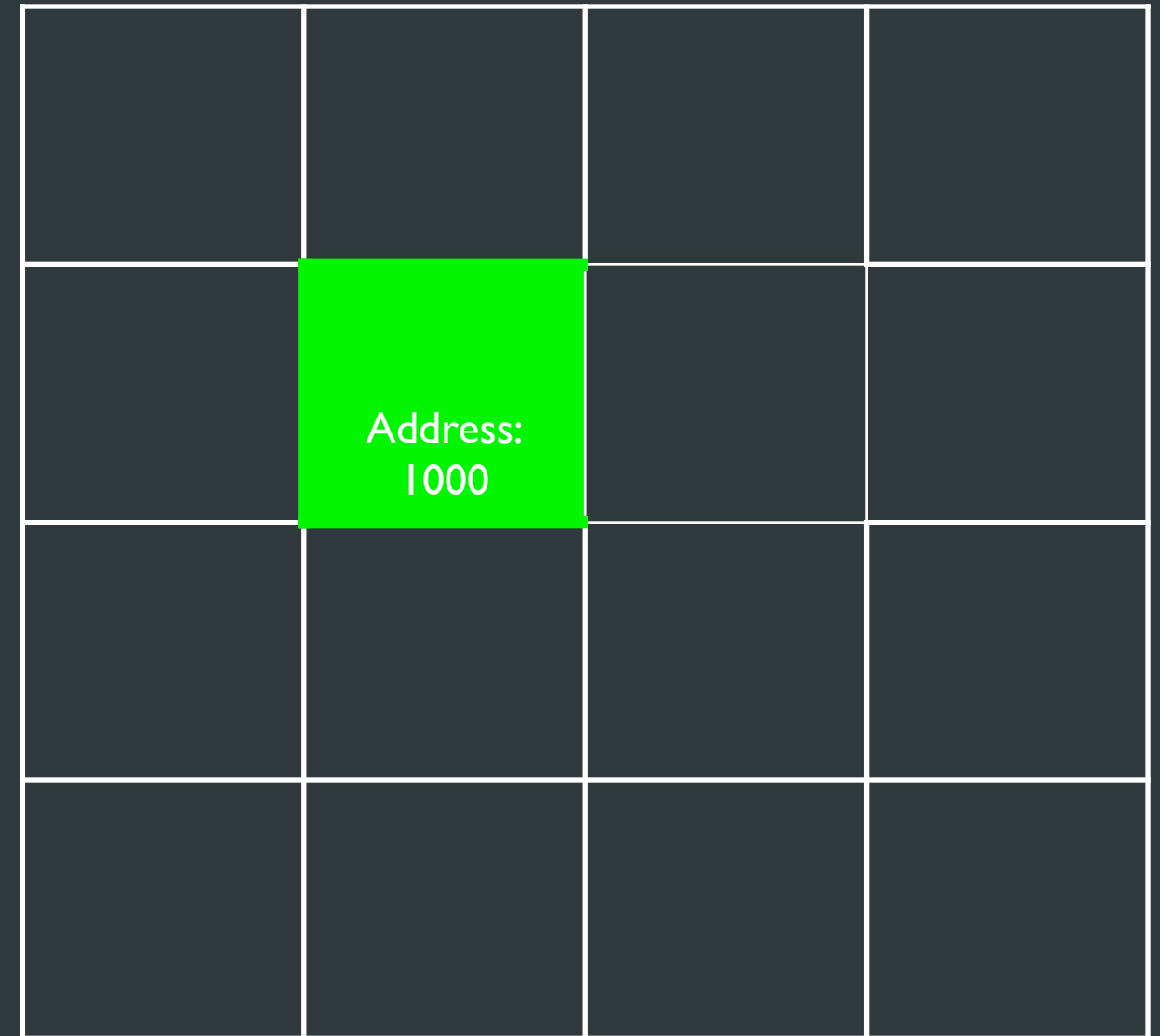
Objective-C Alloc & Init

NSNumber

```
NSNumber *number;
```


NSNumber

```
NSNumber *number;  
number = [NSNumber alloc];
```



NSNumber

```
NSNumber *number;  
number = [NSNumber alloc];  
[number initWithInt:100];
```

	100 Address: 1000		

NSNumber

```
NSNumber *number;  
number = [NSNumber alloc];  
[number initWithInt:100];
```

// equivalent to

```
NSNumber *number = [NSNumber alloc];  
[number initWithInt:100];
```

// equivalent to

```
NSNumber *number = [[NSNumber alloc] initWithInt:100];
```

Objective-C Literals @

Literals - shorthand notation

```
NSNumber *fortyTwo = @42;
```

```
// equivalent to
```

```
[NSNumber numberWithInt:42]
```

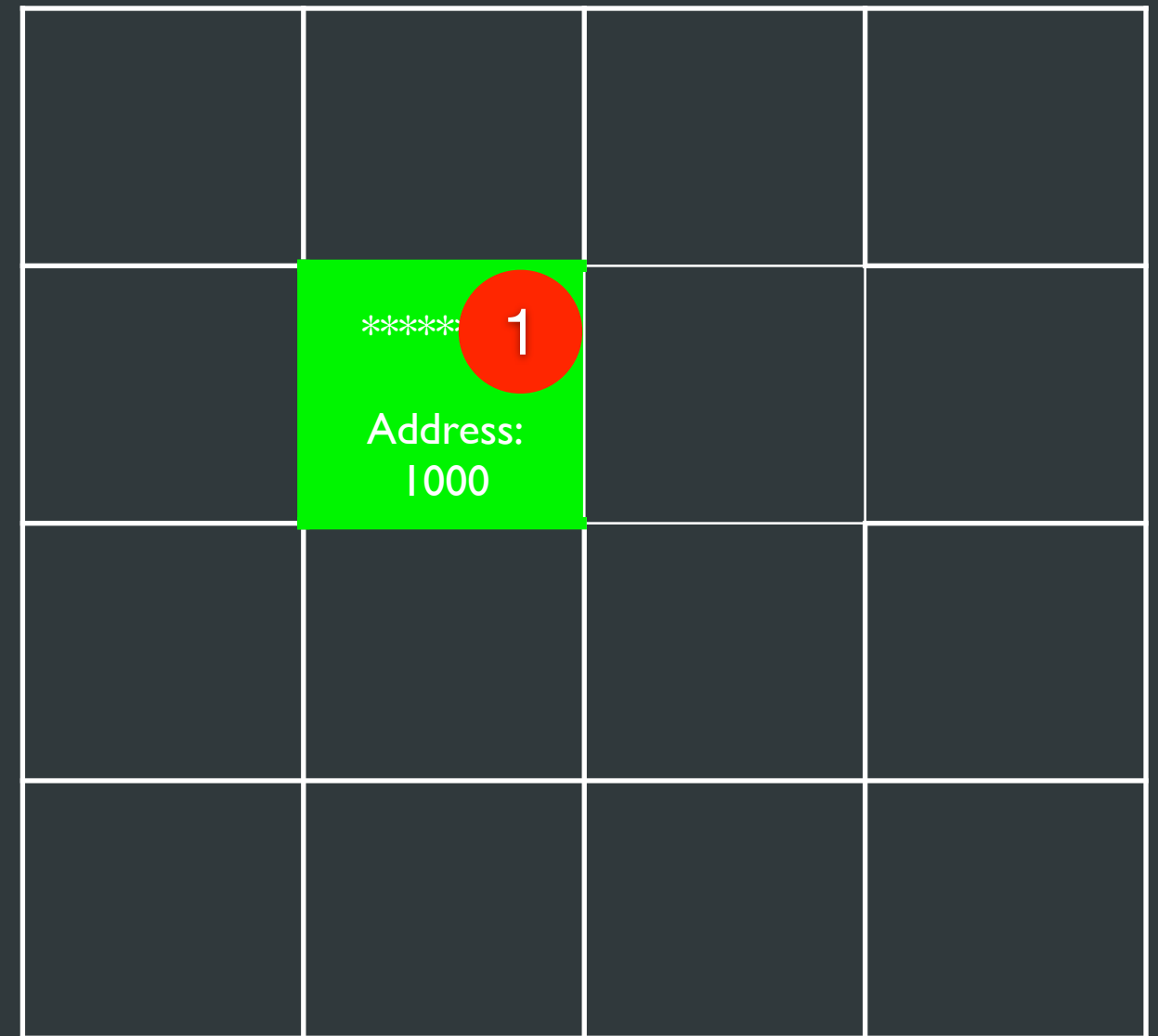
@“Hello”;

```
@“.”; // NSString
@100 // NSNumber
@[] // NSArray
@{} // NSDictionary
```

Property Attributes

Retain Counts

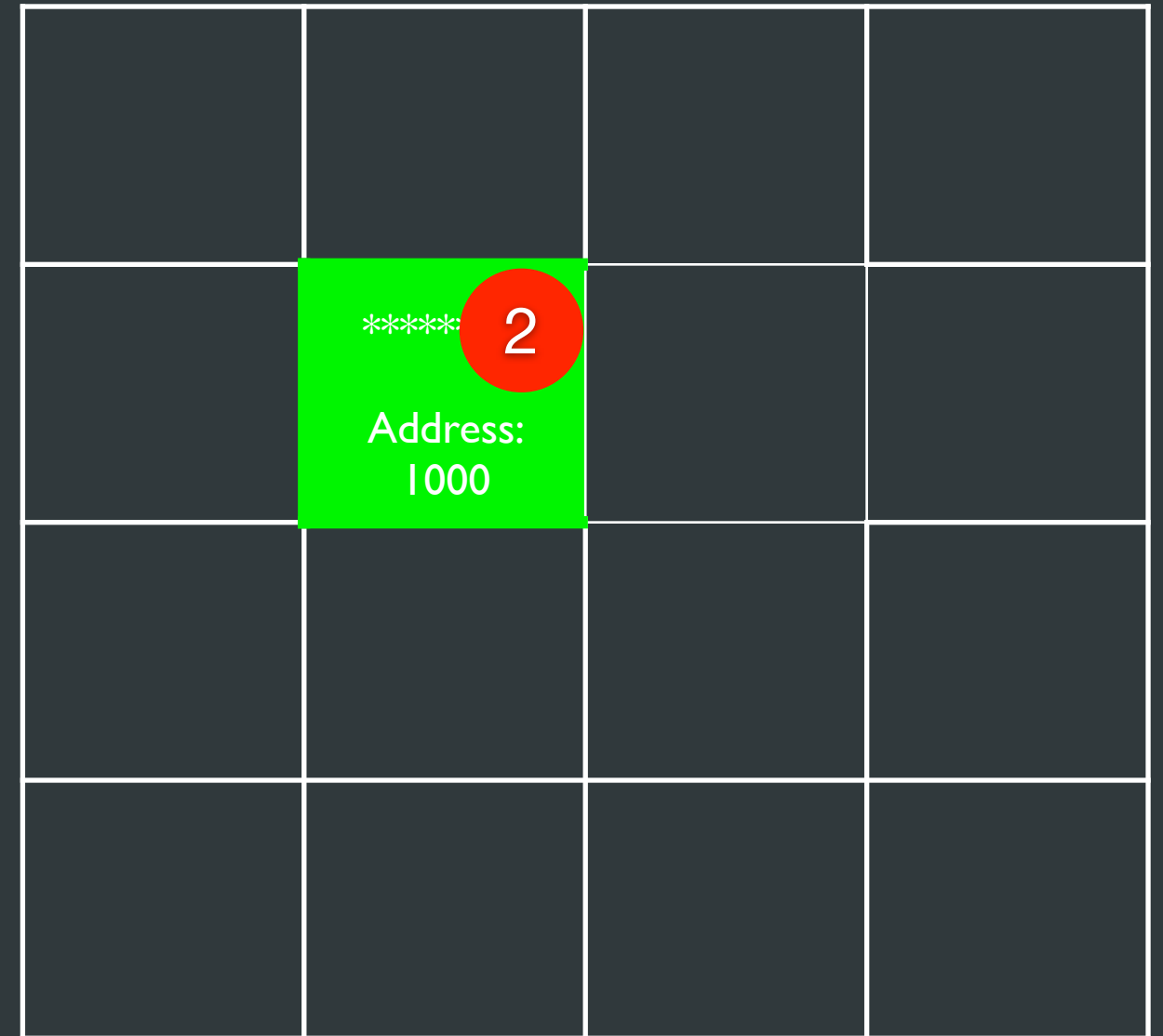
```
Book *book1 = [[Book alloc] init];
```



Retain Counts

```
Book *book1 = [[Book alloc] init];
```

```
Book *book2 = book1;
```

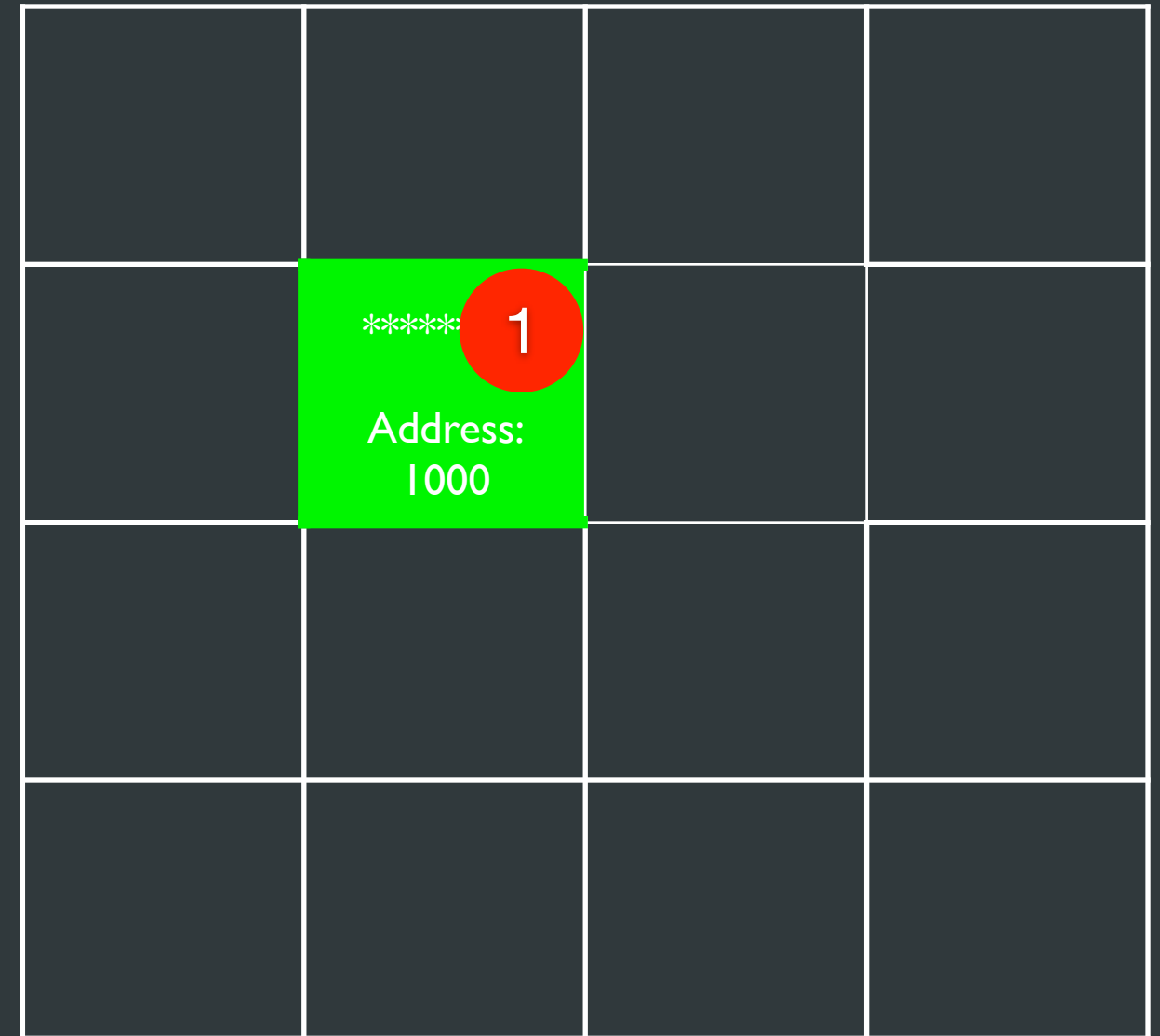


Retain Counts

```
Book *book1 = [[Book alloc] init];
```

```
Book *book2 = book1;
```

```
book2 = nil;
```



Retain Counts

```
Book *book1 = [[Book alloc] init];
```

```
Book *book2 = book1;
```

```
book2 = nil;
```

```
book1 = nil;
```

	0		

Thank You