Variables and Types

The buildings blocks of apps

Paul Solt
iPhoneDev.tv

Coffee

+

=

Paul Solt
iPhoneDev.tv

Coffee

- 17 ml water : 1 g coffee
- 950 ml water / 17 ml/g = ?
- 55.8 g coffee

Paul Solt
iPhoneDev.tv

Variables

Give the CPU something to remember

Paul Solt
iPhoneDev.tv

```c
 // Declare a variable to store water
float water;


// Store the amount of water to use
water = 950;  // milliliters


// Display message
printf("Brew %f milliliters of coffee", water);
```

```
int age = 26;
```

Paul Solt
iPhoneDev.tv

type

```
int age = 26;
```

Paul Solt
iPhoneDev.tv

type name

```
int age = 26;
```

Paul Solt
iPhoneDev.tv

type name expression

```
int age = 26;
```

Paul Solt
iPhoneDev.tv

type name expression

```
int age = 26;
```

assignment operator

Paul Solt
iPhoneDev.tv

```
int age = 26;
```

Paul Solt
iPhoneDev.tv


```
int age = 26;
```

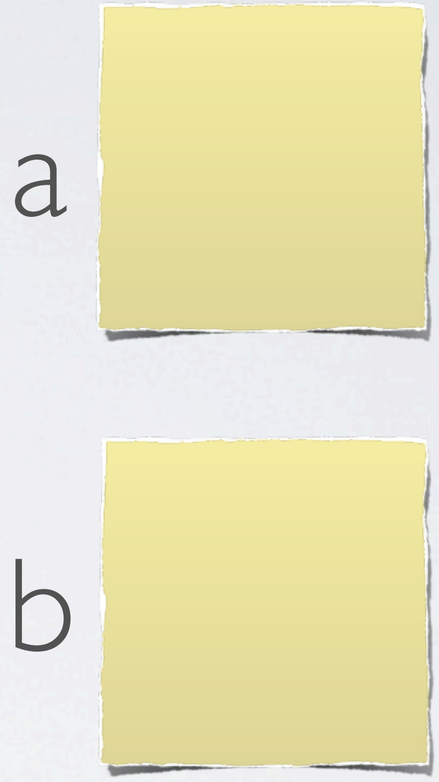
Paul Solt
iPhoneDev.tv

```
int a;
int b;
a = 5;
b = 20;
b = 5 + b;
a = a - b;
```

```
int a;
int b;
a = 5;
b = 20;
b = 5 + b;
a = a - b;
```

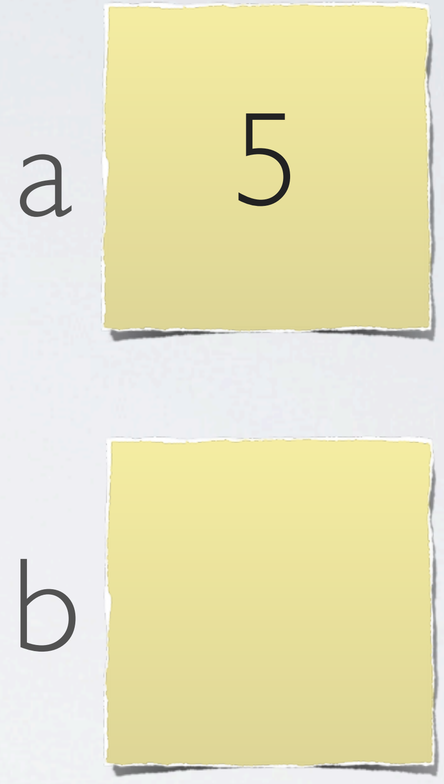
Paul Solt
iPhoneDev.tv

```
int a;
int b;
a = 5;
b = 20;
b = 5 + b;
a = a - b;
```

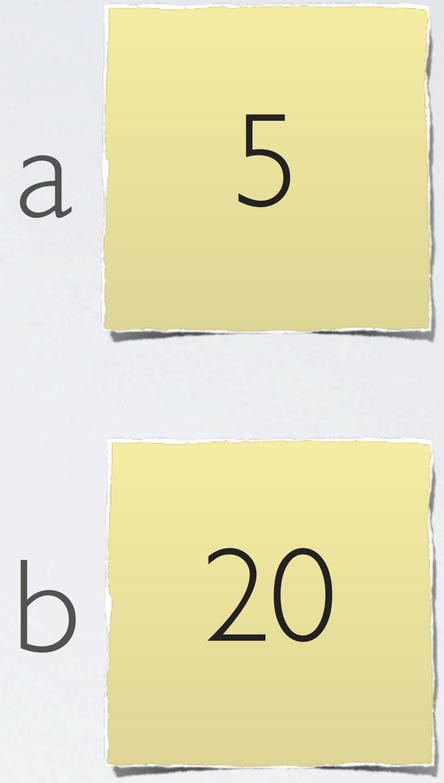


Paul Solt
iPhoneDev.tv

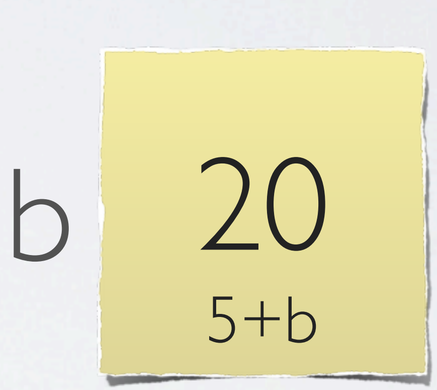
```
int a;
int b;
a = 5;
b = 20;
b = 5 + b;
a = a - b;
```



```
int a;
int b;
a = 5;
b = 20;
b = 5 + b;
a = a - b;
```



```
int a;
int b;
a = 5;
b = 20;
b = 5 + b;
a = a - b;
```

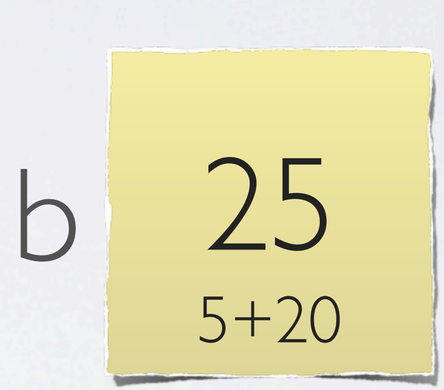


Paul Solt
iPhoneDev.tv

```
int a;
int b;
a = 5;
b = 20;
b = 5 + b;
a = a - b;
```

Paul Solt
iPhoneDev.tv

```
int a;
int b;
a = 5;
b = 20;
b = 5 + b;
a = a - b;
```



Paul Solt
iPhoneDev.tv

```
int a;
int b;
a = 5;
b = 20;
b = 5 + b;
a = a - b;
```

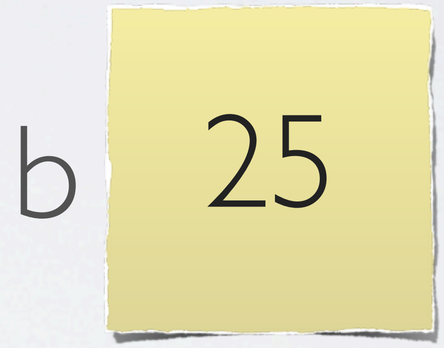
Paul Solt
iPhoneDev.tv

```
int a;
int b;
a = 5;
b = 20;
b = 5 + b;
a = a - b;
```

Paul Solt
iPhoneDev.tv

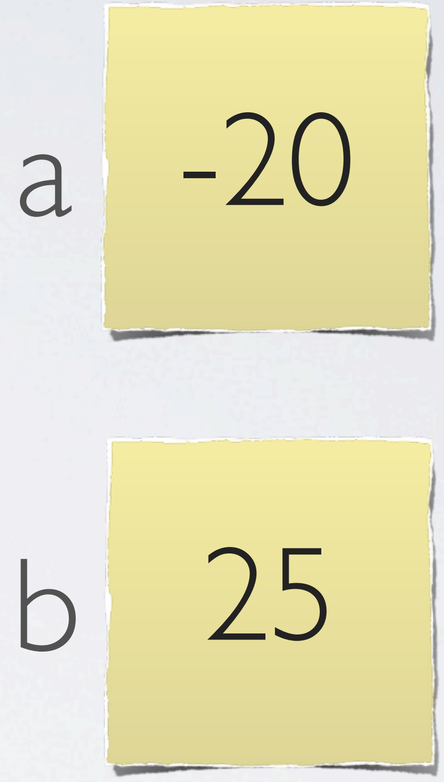
```
int a;
int b;
a = 5;
b = 20;
b = 5 + b;
a = a - b;
```

```
int a;
int b;
a = 5;
b = 20;
b = 5 + b;
a = a - b;
```



Paul Solt
iPhoneDev.tv

```
int a;
int b;
a = 5;
b = 20;
b = 5 + b;
a = a - b;
```



Types

What kind of information are we storing?

Paul Solt
iPhoneDev.tv

- short/int/long: -1,0,1
- float/double: 3.14
- char: ‘a’, ‘b’, ‘c’
- pointers: int * (memory address)
- struct: composition (x, y)

```
short small = 12;
int medium = 2000000;
long large = 9013372684473500;
```

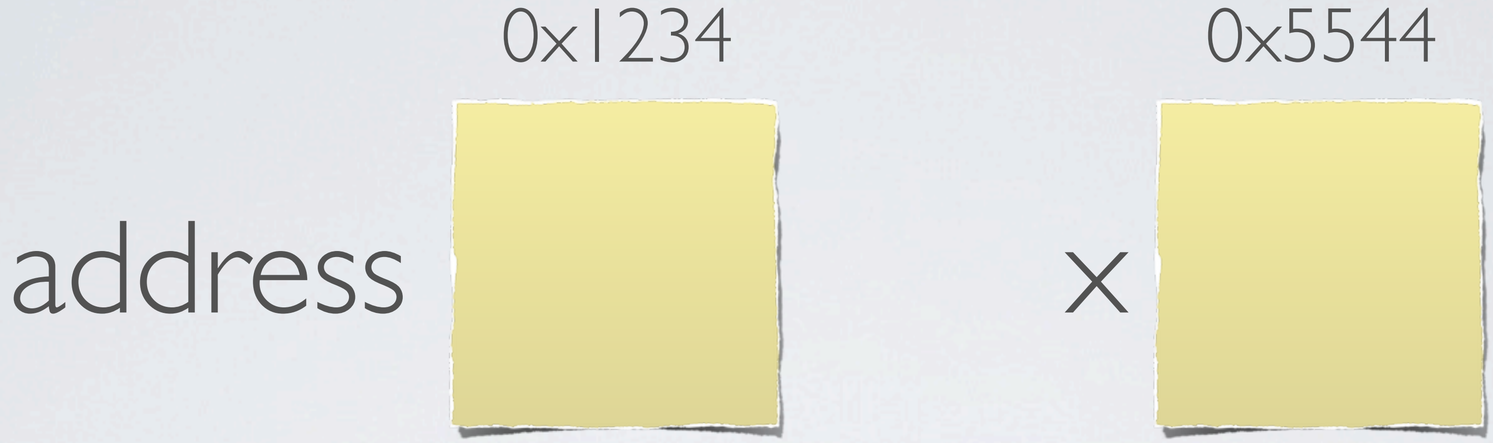
Paul Solt
iPhoneDev.tv

```
float smaller = 3.14;
double larger = 3.14159265359;
```

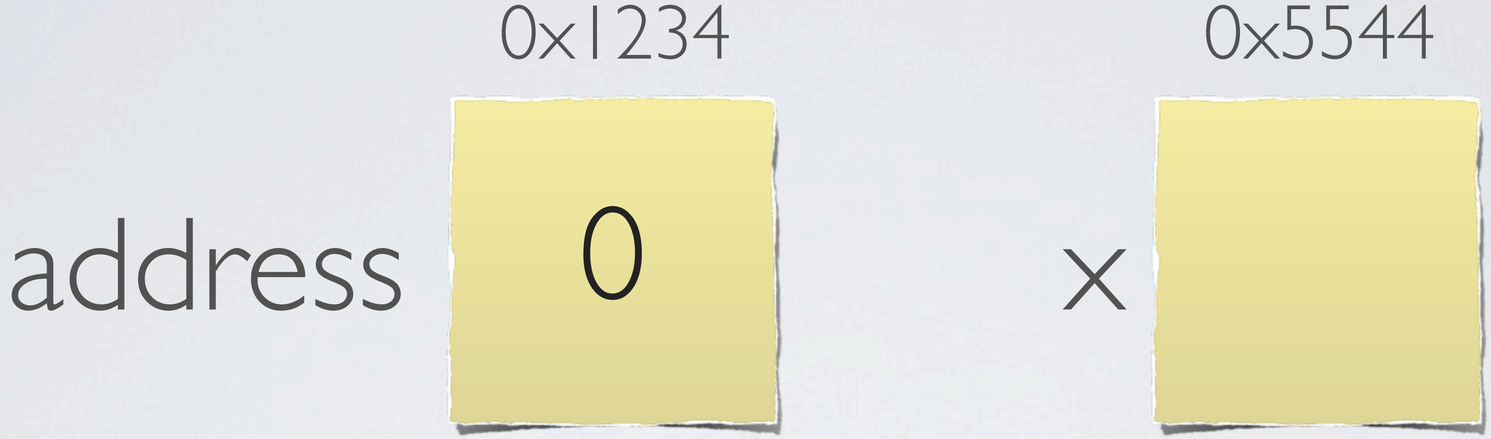
Paul Solt
iPhoneDev.tv

```
char firstLetter = 'a';
char percent = '%';
```

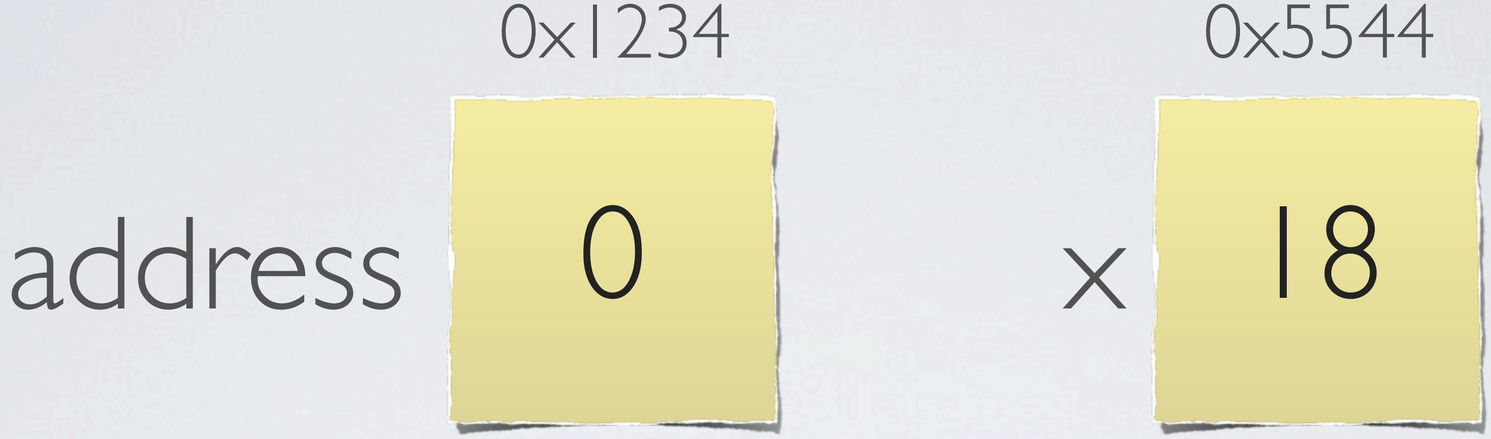
Paul Solt
iPhoneDev.tv




```
int *address = 0;
int x = 18;
address = &x;
*address = 27;
```

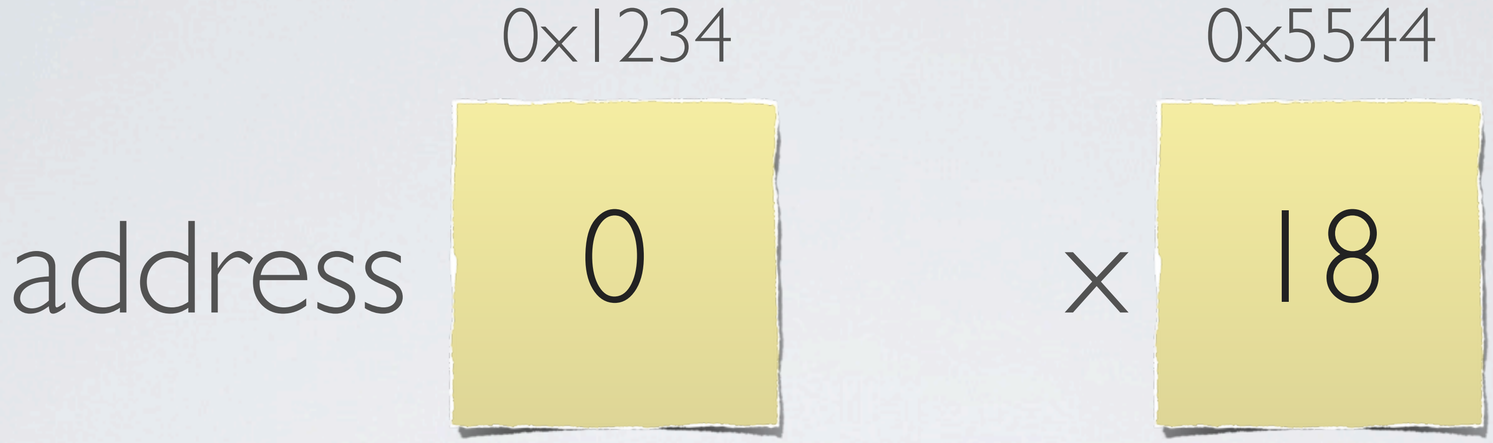



```
int *address = 0;
int x = 18;
address = &x;
*address = 27;
```

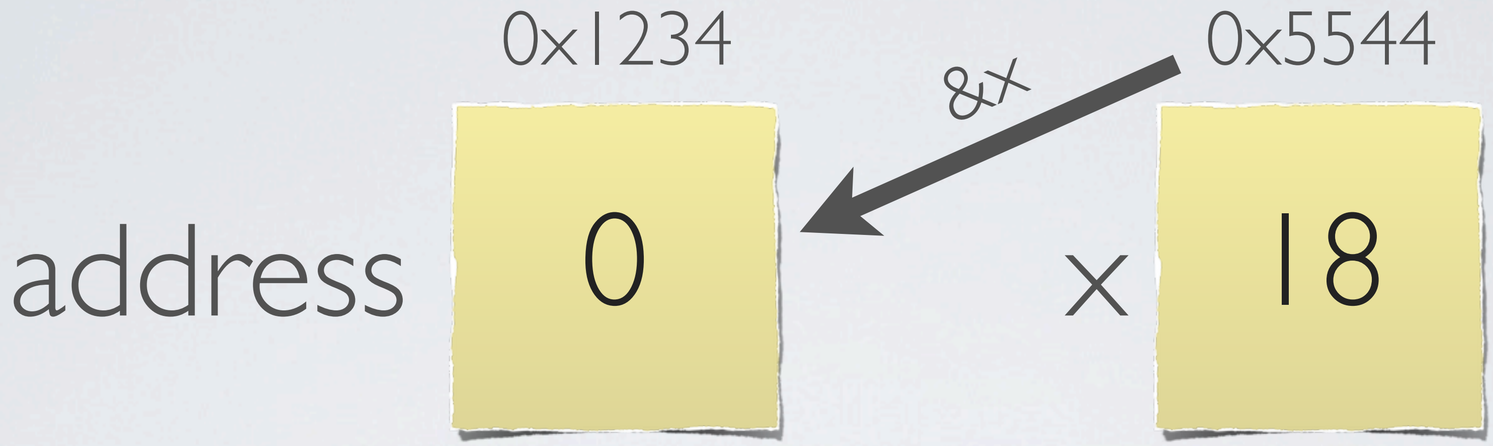



```
int *address = 0;
int x = 18;
address = &x;
*address = 27;
```

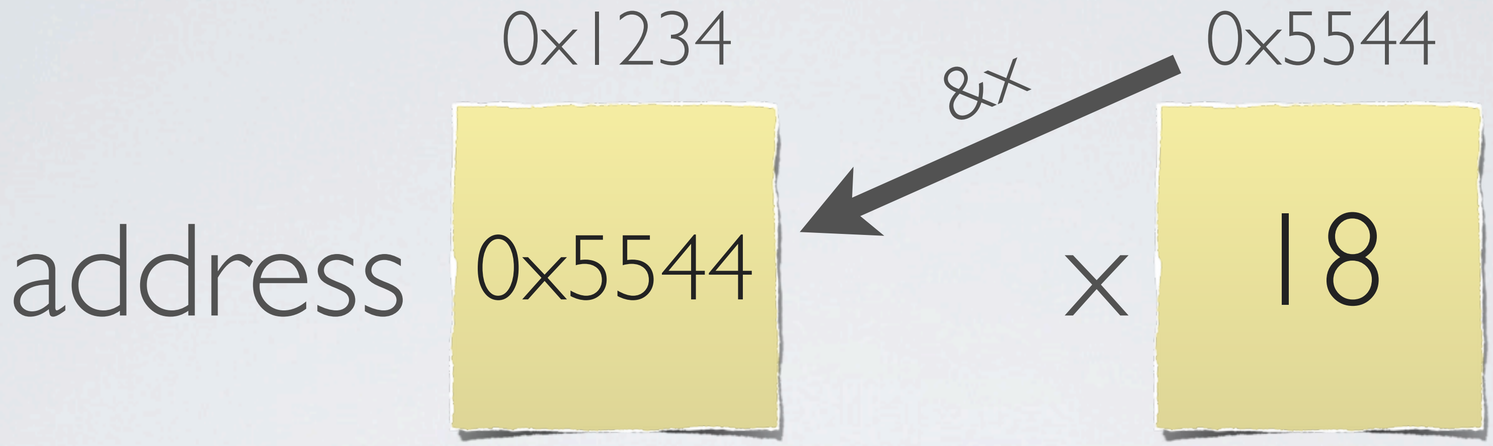
Paul Solt
iPhoneDev.tv



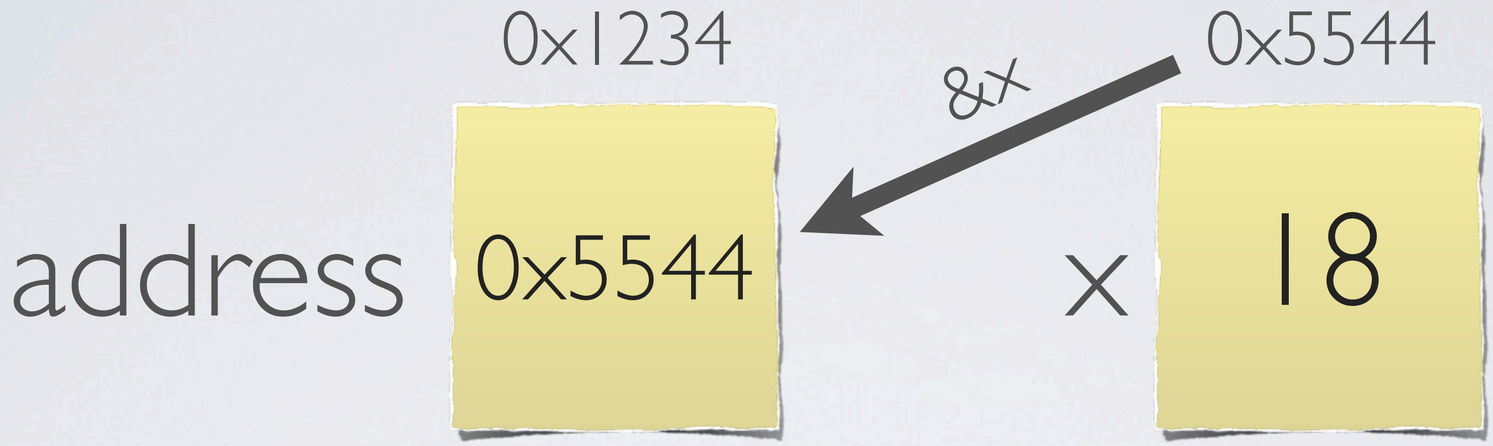

```
int *address = 0;
int x = 18;
address = &x;
*address = 27;
```



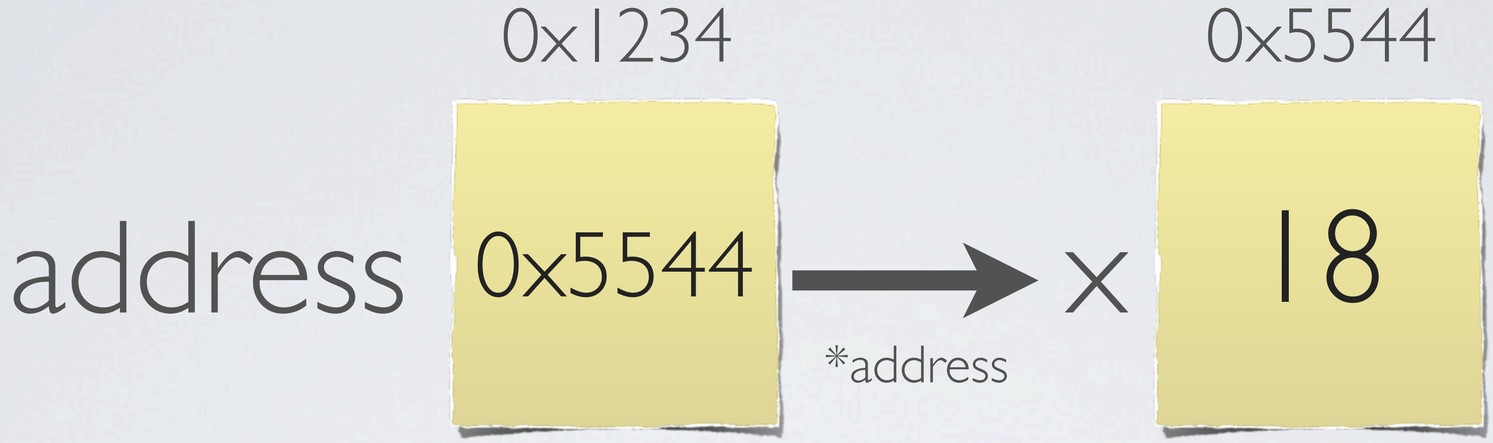

```
int *address = 0;
int x = 18;
address = &x;
*address = 27;
```



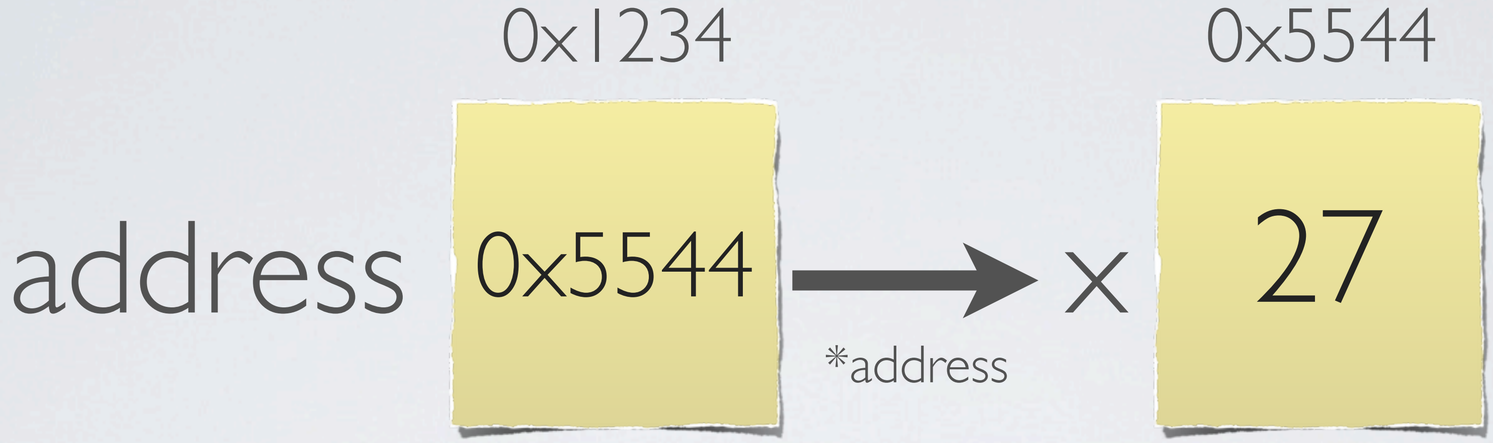

```
int *address = 0;
int x = 18;
address = &x;
*address = 27;
```



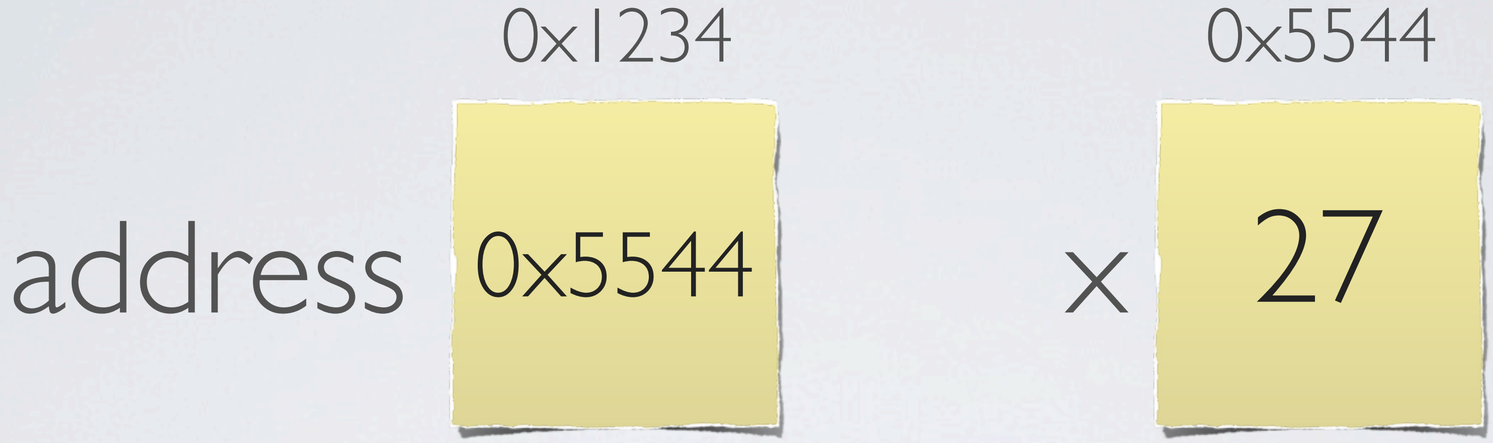

```
int *address = 0;
int x = 18;
address = &x;
*address = 27;
```




```
int *address = 0;
int x = 18;
address = &x;
*address = 27;
```




```
int *address = 0;
int x = 18;
address = &x;
*address = 27;
```




```
int *address = 0;
int x = 18;
address = &x;
*address = 27;
```

```
struct Point {
    int x;
    int y;
};
```

...

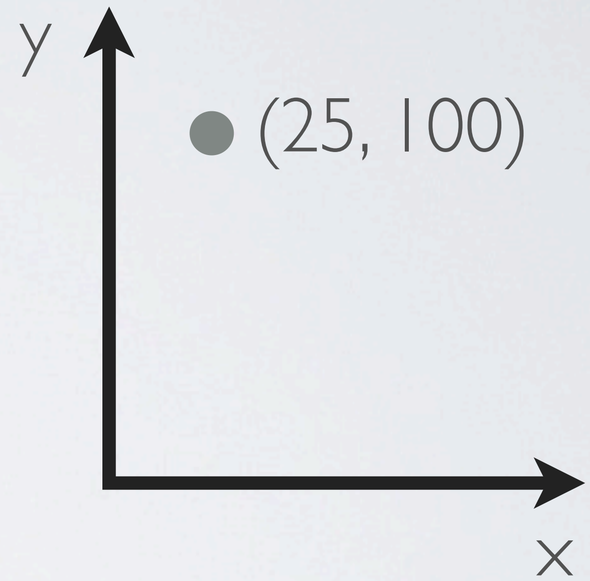
```
struct Point a;
a.x = 25;
a.y = 100;
```

Paul Solt
iPhoneDev.tv

```
struct Point {
    int x;
    int y;
};
```

...

```
struct Point a;
a.x = 25;
a.y = 100;
```



Paul Solt
iPhoneDev.tv

- short/int/long: -1,0,1
- float/double: 3.14
- char: ‘a’, ‘b’, ‘c’
- pointers: int * (memory address)
- struct: composition (x, y)

Paul Solt
iPhoneDev.tv