# The C Programming Language

Kenny Kerr
http://kennykerr.ca

# Pointers and Addresses

- Pointers store addresses of variables
- Pointers are also variables

```
char * ptr_to_char;

void * ptr_to_unknown;

int * ptr_to_nothing = 0;
```

| Address | Type | Value |
|---|---|---|
| 0 | | |
| 1 | | |
| 2 | | |
| … | | |
| 1024 | char | ‘C’ |
| … | | |
| 1040 | char * | 1024 |
| … | | |

# Pointers and Variables

```
int apples = 5;
int * p = &apples;
int oranges = *p + 6;
```

```
int * p;
int oranges = *p + 6;
```
➔ forgot to initialize?
➔ who knows?!

```
int * p = 0;
int oranges = *p + 6;
```
➔ better
➔ boom!

```
int * p = 0;
if (p) { int oranges = *p + 6; }
```
➔ safe!

# Defining Arrays

```
int values[5];
```
➔ who knows but ok

```
int values[] = { 1, 2, 3, 4, 5 };
```
➔ yep, five elements!

```
int values[5] = { 1, 2, 3 };
```
➔ `{ 1, 2, 3, 0, 0 }`

```
char buffer[1024] = { 0 };
```
➔ `{ 0, 0, ... 0, 0 }`

# Arrays and Pointers

```
int values[5];                  ➔ { ?, ?, ?, ?, ? }
values[0] = 1;                  ➔ { 1, ?, ?, ?, ? }
values[4] = values[0];          ➔ { 1, ?, ?, ?, 1 }

int * ptr = values;             ➔ pointing to first element

int * last = &values[4];        ➔ pointing to last

++ptr;                          ➔ pointing to next

ptr += 3;                       ➔ pointing to last
```