Reminders:

- P1 graded, feedback available
- P2 assigned, due Tuesday October 21
- Reading: Chapter 4, Linked Lists

Last class:

- `typedef`
- `void`
- Static variables
- More pointers
  - Pointers as return types
  - Pointers to functions
  - Pointers to pointers
  - Arrays of pointers

Today:

- Classes
- Header Files
- More preprocessor commands
Pointers to Functions

```c
int foo() { ... }

int (*fp) ();
fp = foo; // NOT fp = foo();

fp();
(*fp)();
```

Why would this ever be useful?

Using `typedef` with function pointers

```c
typedef bool (*compareFunction) (int, int);

void sortFunction(int* list, compareFunction cmp) { ... }
```
Classes

Encapsulation

Abstraction

Re-use
Walking through an example

Header files

“Header guards”:

    #ifndef, #define preprocessor commands

public vs. private

using const

Source files

Constructor Basics

Scope resolution operator