CS 162
Intro to CS II
Class Basics
What is const?

• What is const?
  – const int x;
  – void function(const int &x) { ... }
  – void function() const { ... }
  – When would we want to make a member function const? When wouldn’t we?
What is static?

• What is static?
  – Class variable or function
  – static int x; .... Point::x

• Can have a `static const int x=0;`
Static members...

- **Static variables:**
  class math{
  public:
    static double pi;
  }
  double math::pi = 3.14;  //init once outside class
  int main() {
    math m, m1;
    m1.pi=2.0;  //since it isn’t constant, it can change
    cout << m.pi;  //changes for all members
    cout << m1.pi;
    cout << math::pi;
    return 0;
  }
Static members...

- **Static functions:**
  ```cpp
  class math{
  private:
    static const double p=3.14;
  public:
    static const double pi() {
      return p; //can only access static members
    }
  };

  int main() {
    math m, m1;
    cout << m.pi();
    cout << m.pi();
    cout << m1.pi();
    cout << math::pi();
    return 0;
  }
  ```
The Big “Three”

- If dynamic memory allocation in class, then...
  - Destructor
  - Copy Constructor
  - Assignment operator overload, we’ll revisit this...
What is a Destructor?

• Deallocate any member variable dynamically allocated...

• What would this destructor look like then?

```cpp
string::~string() {
    delete [] s; //delete ignores NULL
}
```
What is a copy constructor?

- Used in pass by value
- Returning an object from a function
- Pass the class type to a constructor

```cpp
string:: string(const string &other) {
    len=other.len;
    if(len == 0) s=NULL;
    else {
        s=new char[len];
        for(int i=0; i<len; i++)
            s[i] = other.s[i];
    }
}
```
Demo...