

Today's announcements

- Course website: <https://chara.cs.illinois.edu/cs225/>
- General assistance (ews, svn, etc.) - post to piazza
<https://piazza.com/illinois/spring2016/cs225>
- HW0 hard copy due now
- HW0 electronic copy due 01/25, 11:59pm
- MP1 due 01/26, 11:59pm
- If you are not registered: if you want to get in the class, put your UIN on your HW0 submission.

Structure of a class defn (cont):

```
class sphere{  
public:  
    sphere();  
    sphere(double r);  
    void setRadius(double newRad);  
    double getDiameter() const;  
    ...  
  
private:  
    double theRadius;  
};
```

```
//constructor(s) (next page)  
  
void sphere::setRadius(double newRad) {  
  
}  
  
double sphere::getDiameter() const {  
  
}  
  
...
```

```
int main() {  
    sphere s;  
  
    cout << s.getDiameter() << endl;  
    return 0;  
}
```

Constructors (intro): When you *declare* a sphere, a sphere class constructor is invoked.

Points to remember abt ctors:

1.

2.

3.

```
int main() {
```

```
}
```

```
...
//default constructor
sphere::sphere() {

}

//default constructor, alternative
sphere::sphere()
{

}

//constructor with given radius
sphere::sphere(double r) {

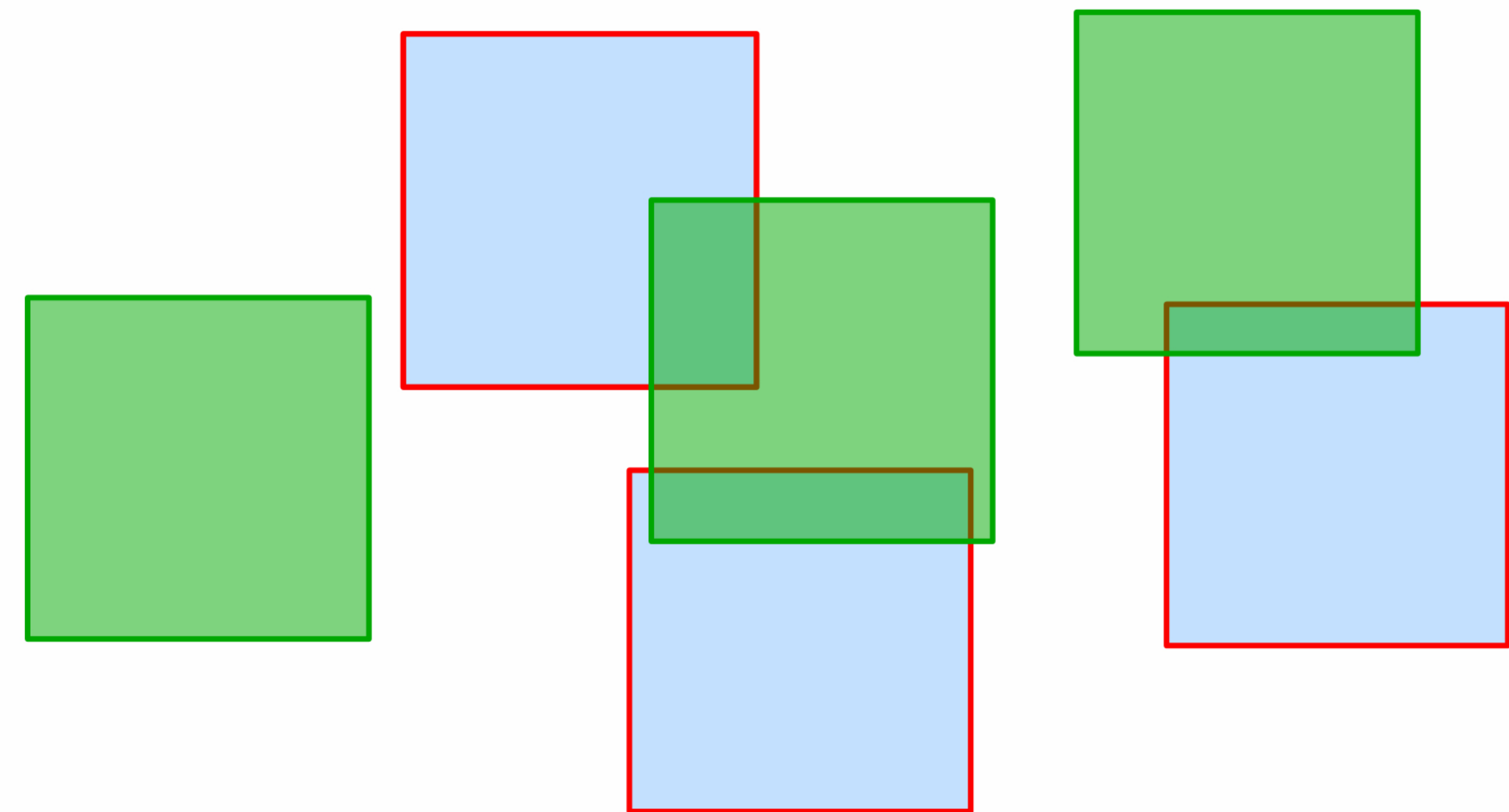
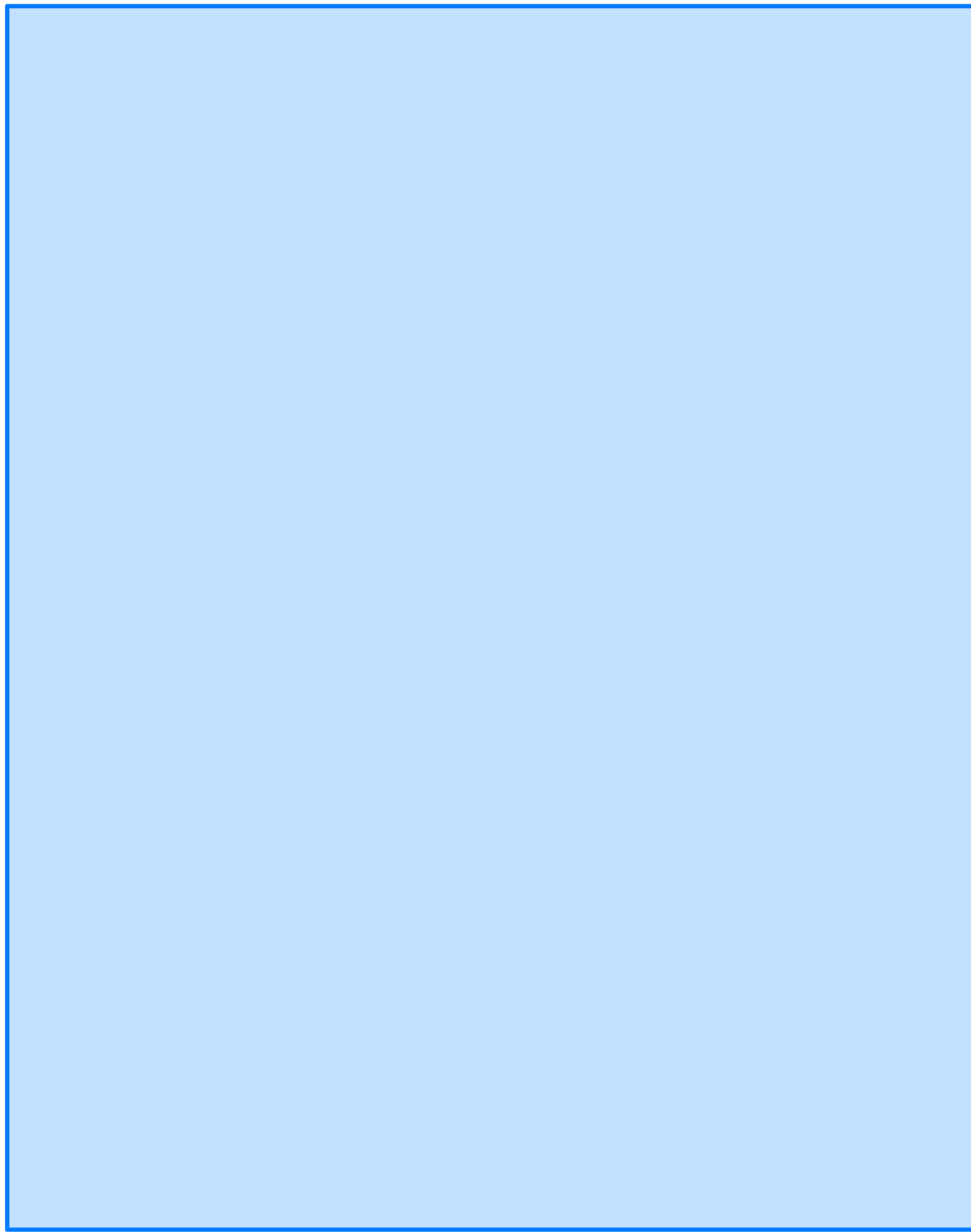
}

...
```

Two Mechanisms of Encapsulation in C++:

1.

2.



Encapsulation: .cpp and .h files

```
class sphere {
public:
    double getDiameter();
    void setRadius(double newRad);
private:
    double radius;
};
```

-- sphere.h All L1 (C/l Abbrev)

```
#include "sphere.h"
void sphere::setRadius(double newRad) {radius = newRad;}
double sphere::getDiameter() {return 2*radius;}
```

-- sphere.cpp All L1 (C++/l Abbrev)

```
#include <iostream>
#include "sphere.h"
using namespace std;

int main() {
    sphere a;
    a.setRadius(10.0);
    cout << "diameter is " << a.getDiameter() << endl;
}
```

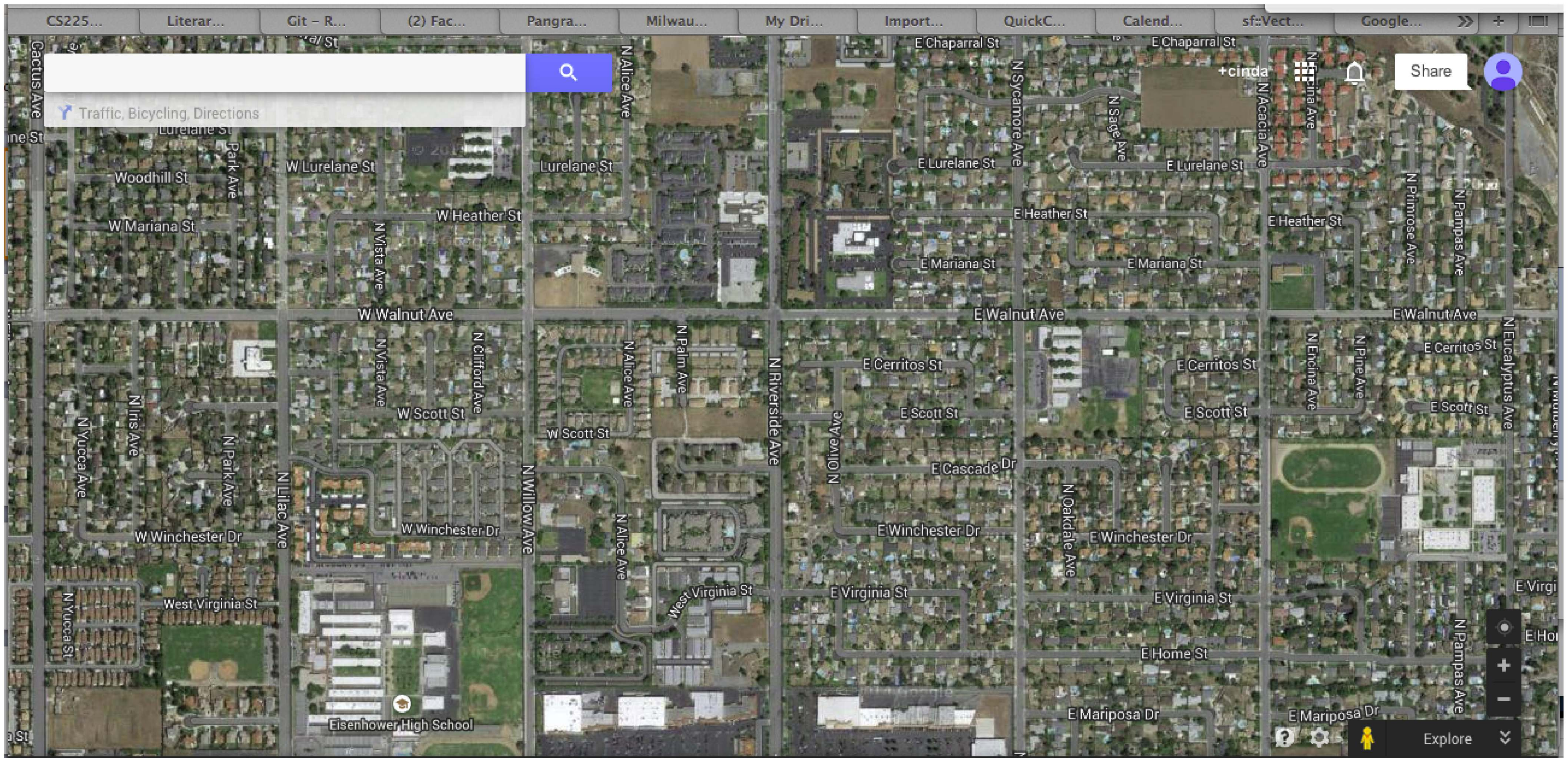
-- main.cpp All L11 (C++/l Abbrev)

To compile:
g++ -c sphere.cpp

To compile:
g++ -c main.cpp

To link into an
executable:
g++ main.o sphere.o

Variables and memory in C++



Variables and memory in C++

Stack memory

[illegible]

Pointers - Intro

```
int x;  
int * p;
```

How do we assign to p?

p =

p =

_____ operator: &

_____ operator: *

Stack memory

loc	name	value	type
a20	x	5	int
a40	p		int *