Towards a more advanced Sphere...
// Create a Sphere with various properties, like color, texture, and others.
#include "Sphere.h"

int main() {
    cs225::Sphere s(10);
    s.addProperty("Red");
    s.addProperty("Rubber");
    return 0;
}
#ifndef SPHERE_H
#define SPHERE_H

namespace cs225 {
    class Sphere {
        public:
            Sphere();
            Sphere(double r);
            Sphere(const Sphere &s);

            // ...
        
        private:
            double r_; 
    };
}

#endif
```
#ifndef SPHERE_H
#define SPHERE_H

namespace cs225 {
    class Sphere {
    public:
        Sphere();
        Sphere(double r);
        Sphere(const Sphere &s);
        // ...
    private:
        double r_;
    }
}
#endif
```

```
#include "sphere.h"

namespace cs225 {
    // ...
}
```
```cpp
#ifndef SPHERE_H
#define SPHERE_H

namespace cs225 {

class Sphere {
public:
    Sphere();
    Sphere(double r);
    Sphere(const Sphere &s);

    // ...

private:
    double r_;  // ...

};

}  // namespace cs225

#endif

#include "sphere.h"

namespace cs225 {

// ...

}  // namespace cs225
```
```cpp
#ifndef SPHERE_H
#define SPHERE_H

namespace cs225 {
  class Sphere {
    public:
      Sphere();
      Sphere(double r);
      Sphere(const Sphere &s);
      // ...
  public:
    private:
      double r_;  // ...
    private:
      // ...
  };
}
#endif
```

```cpp
#include "sphere.h"

namespace cs225 {
  // ...
}
```
Labs

Every lab is worth 10 points.

**Attendance:** 4 points (40%)
- Work on your lab for the full lab section, **and/or**
- *Anytime after the intro*, show your TA a completely passing test suite when you’ve finished the lab!
- Only will get credit in your official lab section

**Correctness:** 6 points (60%)
- Complete and submit your lab by Sunday @ 11:59pm
Labs

Lab points contribute 100 points to the base 1000 points in this course.

lab_intro + lab_debug ➞ 20 points
13 labs remain * 6 points ➞ 78 points
= 98 points for correctness

13 labs remain * 4 points ➞ 52 attendance points
(50 of those are EC!)
Inheritance
#ifndef REDBALL_H
#define REDBALL_H

#include "Sphere.h"

public RedBall
   public:

private:

} // RedBall
#endif

#include "RedBall.h"
A

[Constructor]:

[Inherits]:

[Inherits]:

[Inherits]:

[Inherits]:
Sphere.cpp

```
Sphere::print_1() {
    cout << "Sphere" << endl;
}

Sphere::print_2() {
    cout << "Sphere" << endl;
}

virtual Sphere::print_3() {
    cout << "Sphere" << endl;
}

virtual Sphere::print_4() {
    cout << "Sphere" << endl;
}

// In .h file:
virtual Sphere::print_5() = 0;
```

RedBall.cpp

```
// No print_1() in RedBall.cpp

RedBall::print_2() {
    cout << "Ball" << endl;
}

RedBall::print_4() {
    cout << "Ball" << endl;
}

// No print_3() in RedBall.cpp

RedBall::print_4() {
    cout << "Ball" << endl;
}

RedBall::print_5() {
    cout << "Ball" << endl;
}
```
### Runtime of Virtual Functions

<table>
<thead>
<tr>
<th>Sphere obj;</th>
<th>RedBall obj;</th>
<th>RedBall r; Sphere &amp;s = r;</th>
</tr>
</thead>
<tbody>
<tr>
<td>obj.print_1();</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>obj.print_2();</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>obj.print_3();</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>obj.print_4();</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>obj.print_5();</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
Abstract Class:

[Requirement]:

[Syntax]:

[Therefore]:
```cpp
class Sphere {
public:
    virtual Sphere();
}

class Ball : public Sphere {
public:
    ________________________________;
};
```
```cpp
class Sphere {
public:
    virtual ~Sphere();
};

class Ball : public Sphere {
public:
    ____________________________;
};
```
Call Order – How are derived classes created?
Call Order – How are derived classes destroyed?
Exam 1 is happening now
Exam 2 registration is available (programming exam)
More Info: https://courses.engr.illinois.edu/cs225/fa2017/exams/

lab_inheritance
Due: Sunday, Sept. 17 (11:59pm)

MP2 is out – Early Deadline Monday, Sept. 18
Up to +7 Extra Credit for Early Submission

POTD
Every Monday-Friday – Worth +1 Extra Credit /problem (up to +40 total)