CS 225

Data Structures
```cpp
#include <iostream>
using namespace std;

int main() {
    int first = 42;
    int arr[6];
    cout << &first << endl;
    cout << &(arr[0]) << endl;
    cout << &(arr[1]) << endl;
    cout << &(arr[2]) << endl;
    return 0;
}
```
#include <iostream>
using namespace std;

int main() {
    int first = 42;
    int arr[6];

    cout << &first << endl;
    cout << &arr[0] << endl;
    cout << &arr[1] << endl;
    cout << &arr[2] << endl;

    return 0;
}
```cpp
#include <iostream>
using namespace std;

int main() {
    int first = 42;
    int arr[6];

    cout << &first << endl;
    cout << &arr[0] << endl;
    cout << &arr[1] << endl;
    cout << &arr[2] << endl;
    return 0;
}
```
Exam 1

In the Computer Based Testing Center (CBTF)
https://cbtf.engr.illinois.edu/

Covers material through last week.
No coding!
Practice exam will be available.
These have started!
https://prairielearn.engr.illinois.edu/

You will have 24 hours from when they are released. Usually 8am each day.
Have fun!
#include "sphere.h"
using namespace cs225;

int main() {
    int *p = new int;
    int *s = new Sphere(10);
    return 0;
}
```cpp
#include "sphere.h"
using namespace cs225;

int main() {
    Sphere *s1 = new Sphere();
    Sphere *s2 = s1;
    s2->setRadius(10);
    return 0;
}
```
Heap Memory Lifecycle
Accessing Memory and Data

\& S

Sphere s
Accessing Memory and Data

*p

Sphere*  ptr
```cpp
#include <iostream>
using namespace std;

int main() {
    int *p, *q;
    p = new int;
    q = p;
    *q = 8;
    cout << *p << endl;

    q = new int;
    *q = 9;
    cout << *p << endl;
    cout << *q << endl;
    return 0;
}
```
#include <iostream>

using namespace std;

int main() {
    int *x;
    int size = 3;
    x = new int[size];
    for (int i = 0; i < size; i++) {
        x[i] = i + 3;
    }
    delete[] x;
}
#include <iostream>

using namespace std;

int main() {
    int *x = new int;
    int &y = *x;

    y = 4;

    cout << &x << endl;
    cout << x << endl;
    cout << *x << endl;
    cout << &y << endl;
    cout << y << endl;
    cout << *y << endl;
}

Reference Variable
/*
 * Creates a new sphere that contains the exact volume
 * of the two input spheres.
 */

Sphere joinSpheres(Sphere s1, Sphere s2) {
    double totalVolume = s1.getVolume() + s2.getVolume();
    double newRadius = std::pow(
        (3.0 * totalVolume) / (4.0 * 3.141592654),
        1.0/3.0
    );
    Sphere result(newRadius);
    return result;
}
Register for Exam 1 (CBTF)
Details on the course website!

Every day, work on the POTDs
Available on PrairieLearn, every weekday!

Finish MP1
Due: Monday, Sept. 11\(^{\text{th}}\) (11:59pm)

Attend lab and complete lab_debug
Due: Sunday, Sept. 10\(^{\text{th}}\) (11:59pm)