CS 225

#4: Heap Memory

January 24, 2018 · *Wade Fagen-Ulmschneider*

Puzzle from Monday

puzzle.cpp

```
 4  Sphere *CreateUnitSphere() {
 5    Sphere s(1);
 6    return &s;
 7  }
 8
 9  int main() {
10    Sphere *s = CreateUnitSphere();
11    someOtherFunction();
12    double r = s->getRadius();
13    double v = s->getVolume();
14    return 0;
15  }
```

Takeaway:

Heap Memory:

As programmers, we can use heap memory in cases where the *lifecycle of the variable exceeds the lifecycle of the function*.

1. The only way to create heap memory is with the use of the **new** keyword. Using **new** will:
   -
   -
   -

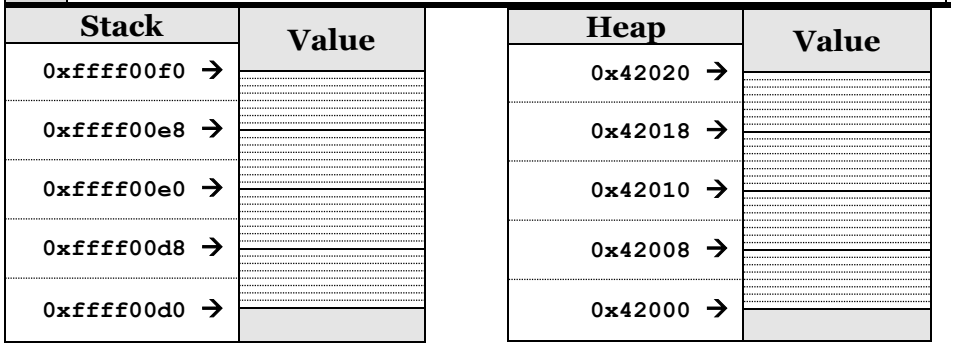
2. The only way to free heap memory is with the use of the **delete** keyword. Using **delete** will:
   -
   -

3. Memory is never automatically reclaimed, even if it goes out of scope. Any memory lost, but not freed, is considered to be "leaked memory".

---

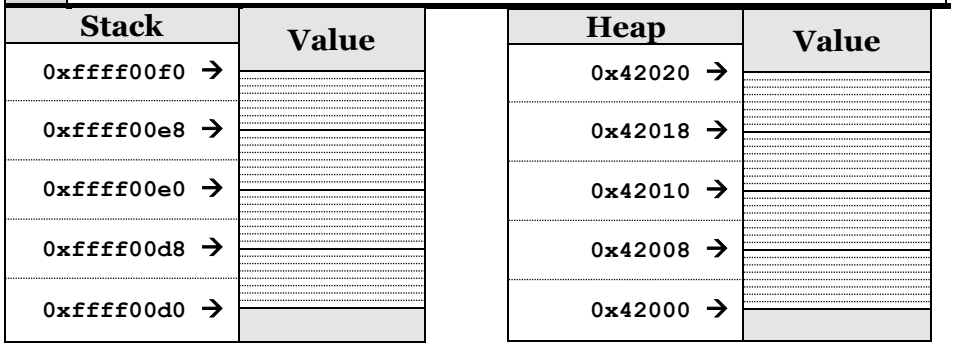
heap1.cpp

```
 4  int main() {
 5    int *p = new int;
 6    Sphere *s = new Sphere(10);
 7
 8    return 0;
 9  }
```



heap2.cpp

```
 4  int main() {
 5    Sphere *s1 = new Sphere();
 6    Sphere *s2 = s1;
 7    s2->setRadius( 10 );
 8    delete s2;
 9    delete s1;
10    return 0;
11  }
```



Copying Memory – Deep Copy vs. Shallow Copy

copy.cpp

```
 5  int  i =  2,  j =  4,  k =  8;
 6  int *p = &i, *q = &j, *r = &k;
 7
 8  k = i;
 9  cout << i << j << k << *p << *q << *r << endl;
10
11  p = q;
12  cout << i << j << k << *p << *q << *r << endl;
13
14  *q = *r;
15  cout << i << j << k << *p << *q << *r << endl;
```

Consider how each assignment operator changes the data:

	Type of LHS	Type of RHS	Data Changed?
Line 8-9			
	i =	**j** =	**k** =
	p =	**q** =	**r** =
Line 11-12			
	i =	**j** =	**k** =
	p =	**q** =	**r** =
Line 14-15			
	i =	**j** =	**k** =
	p =	**q** =	**r** =

heap-puzzle1.cpp

```
 5  int *x = new int;
 6  int &y = *x;
 7
 8  y = 4;
 9
10  cout << &x << endl;
11  cout << x << endl;
12  cout << *x << endl;
13
14  cout << &y << endl;
15  cout << y << endl;
16  cout << *y << endl;
```

heap-puzzle2.cpp

```
 5  int *p, *q;
 6  p = new int;
 7  q = p;
 8  *q = 8;
 9  cout << *p << endl;
10
11  q = new int;
12  *q = 9;
13  cout << *p << endl;
14  cout << *q << endl;
```

heap-puzzle3.cpp

```
 5  int *x;
 6  int size = 3;
 7
 8  x = new int[size];
 9
10  for (int i = 0; i < size; i++) {
11    x[i] = i + 3;
12  }
13
14  delete[] x;
```

joinSpheres.cpp

```
11  /*
12   * Creates a new sphere that contains the exact volume
13   * of the two input spheres.
14   */
15  Sphere joinSpheres(Sphere   s1, Sphere   s2) {
16    double totalVolume = s1.getVolume() + s2.getVolume();
17
18    double newRadius = std::pow(
19      (3.0 * totalVolume) / (4.0 * 3.141592654),
20      1.0/3.0
21    );
22
23    Sphere result(newRadius);
24
25    return  result;
26  }
```

CS 225 – Things To Be Doing:

1. Exam 0 is ongoing – ensure you're signed up for it!
2. Finish up MP1 – Due Monday, Jan. 29 at 11:59pm
3. Complete lab_debug this week in lab sections (due Sunday)
4. POTDs are released daily, worth +1 extra credit point! ☺