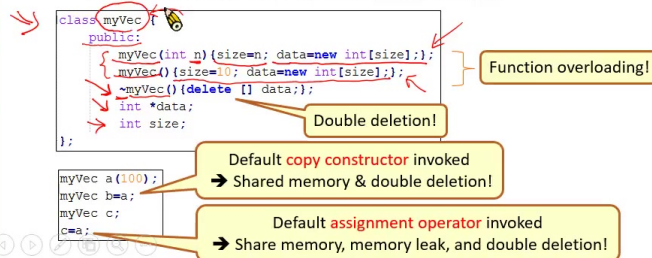


## Memory Allocation in Classes

- Common programming errors when using "new" for memory allocation in classes
  - Default **copy constructor** and **assignment operator** are based on "shallow copy", which leads to errors easily.
  - We need to design our own constructor/operator.



```

class myVec {
public:
    myVec(int n) {size=n; data=new int[size];};
    myVec() {size=10; data=new int[size];};
    ~myVec() {delete [] data;};
    int *data;
    int size;
};

myVec a(100);
myVec b=a;
myVec c;
c=a;
    
```

Function overloading!

Double deletion!

Default copy constructor invoked  
→ Shared memory & double deletion!

Default assignment operator invoked  
→ Share memory, memory leak, and double deletion!

2

(3:36)

by Oliver Chang

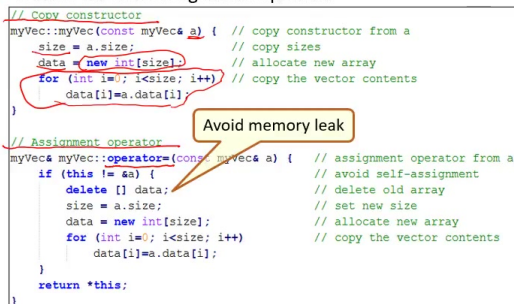
class ->private

struct ->public

If a class allocates memory via "new", then need to provide new copy instructor & assignment operator to allocate new memroy for the created copy.

## How to Fix "Shallow Copy"?

- To fix the problem of "shallow copy"
  - Define our own copy constructor
  - Define our own assignment operator



```

// Copy constructor
myVec::myVec(const myVec& a) { // copy constructor from a
    size = a.size;           // copy sizes
    data = new int[size];     // allocate new array
    for (int i=0; i<size; i++) // copy the vector contents
        data[i]=a.data[i];
}

// Assignment operator
myVec& myVec::operator=(const myVec& a) { // assignment operator from a
    if (this != &a) { // avoid self-assignment
        delete [] data; // delete old array
        size = a.size; // set new size
        data = new int[size]; // allocate new array
        for (int i=0; i<size; i++) // copy the vector contents
            data[i]=a.data[i];
    }
    return *this;
}
    
```

Avoid memory leak

3

(10:13)

by Oliver Chang

Every class that allocates its own objects using new should:

Define a destructor to free any allocated objects.

Define a copy constructor, which allocates its own new memory and copies the contents of member variables.

Define an assignment operator, which deallocates old storage, allocates new storage, and copies all member variables.