

Classes and Traditional Pointers

Simple Class with Pointer

```
class WithPointer {  
    char *p;  
    int n;  
public:  
    WithPointer(int n) : n(n) {  
        p = new char[n];  
    }  
    ~WithPointer() { delete [] p; }  
};
```

```
WithPointer a(5);
```

```
// Create an instance of the class
```

Copy Constructor with deep copy

// Copy constructor which makes a deep copy

```
WithPointer(const WithPointer& other) : n(other.n) {
```

```
    p = new char[n];           // Acquire our own version of the pointer
```

```
    for (int i = 0; i < n; ++i) { // Initialize it from the copied instance's pointer data
```

```
        p[i] = other.p[i];
```

```
    }
```

```
}
```

Assignment Operator with deep copy

// Assignment operator which makes a deep copy

```
WithPointer& operator =(const WithPointer& other) {  
    if (this != &other) {           // Check for self-assignment  
        delete [] p;                // Perform the same code as the destructor  
        n = other.n;                 // Perform the same code as the copy constructor  
        p = new char[n];  
        for (int i = 0; i < n; ++i) {  
            p[i] = other.p[i];  
        }  
    }  
    return *this;                    // Return a non-const reference to this instance  
}
```