The Big 5
Classes – No Internal "new"

• Member variables

• Methods

• Constructor

• Rule of Zero
Case Study: StringHolder

class StringHolder {
    std::string *string_;  
public:
    StringHolder() : string_(nullptr) {};
    StringHolder(const char *initial_string);

    const char* c_str() const noexcept;
    void ChangeString(const char *input_string);
};
Classes – Containers Use "new"

• Rule of Zero won't work

• Deep vs Shallow Copy
Rule of Three – Copy Assignment Operator

StringHolder& operator=(const StringHolder &source)
Rule of Three - Copy Constructor

StringHolder(const StringHolder &source)
Rule of Three - Destructor

~StringHolder()
Rule of Three vs Rule of Five

• Rule of five (move semantics new with C++11)
  • Move Constructor
  • Move Assignment Operator

• rvalue reference &&
  • Can bind to a temporary (rvalue)
Rule of Three - Move Constructor

StringHolder(StringHolder&& source)
Rule of Three – Move Assignment Operator

```cpp
StringHolder& operator=(StringHolder&& source)
```
Lists?