

# Writing a Class using RAII Exercises

- Explain how using RAII ensures the strong exception guarantee for
  - Constructor
  - Copy constructor
  - Assignment Operator
  - Destructor

- Explain why the assignment operator, as usually written for an RAI class, offers only the basic exception guarantee

- Explain in detail how the function shown below works
- Explain why this offers the strong exception guarantee
- Describe some of the other advantages this implementation has over the version without a temporary

```
BufferManager& operator =(const BufferManager& other) {  
    BufferManager temp(other); // Create copy of other  
    swap(*this, temp);         // Replace our data with temp's data  
    return *this;  
}                               // Destroy temp
```

- Implement the "rule of three" operators for the class shown below using the RAII idiom

```
class BufferManager {  
    private:  
        int size;  
        char *buffer;  
        ....  
};
```