

ADVANCED PROGRAMMING

SIR FAHAD AHMED SATTI

ASSIGNMENT 2

SMART STRINGS

DOCUMENTATION



**INTRODUCTION**

4

.CPP

Files

* Copied\_Pointers
* Owned\_Pointers
* COW\_with\_Reference\_Counting
* COW\_with\_Reference\_Linking

**HOW TO RUN**

Separate .CPP file for each implementation.

Classes String{} and String\_Buffer{} and the main() function are in the same file for each implementation.

In every file, main() function demonstrates the implementation.

Run each file separately with a C++ compiler OR include 1 file at a time in Visual Studio project and run the project.

**APPROACH**

**Copied Pointers**

When the copy constructor is called, it deep copies the contents of the passed object to the new one. The two different strings have the same contents, stored in separate memory locations.

**Owned Pointers**

The String{} class has a StringBuffer\* member that points to the object in which a char\* points to the actual string. The String{} class has a ‘bool’ variable that tracks ownership. Copied strings contain pointers to the same StringBuffer. When a String copies another String’s contents, the former one gains ownership of the StringBuffer and it becomes its responsibility to release the memory when it goes out of scope.

**Copy-on-Write with Reference Counting**

The StringBuffer{} class has an ‘int’ variable that keeps track of the reference count. Functions are provided to increment or decrement the reference count. When a String gains or loses the pointer to a StringBuffer object, it modifies the count. When reference count becomes 0, the object is deleted.

Copy-on-Write is implemented by creating a new StringBuffer if the reference count is greater than 1, and removing 1 from the previous one’s reference count.

**Copy-on-Write with Reference Linking**

A circular doubly linked list of Strings is maintained instead of a reference count. The String{} class has ‘next’ and ‘previous’ pointers. When a String is copied, it is added to the circular doubly linked list. When a string loses the StringBuffer object, it is removed from the list. If only one node is left, the StringBuffer is deleted.