**String\_Class\_C++:**Logic of methods of String class.

* String();//default const
* String(char \*);//paramtersied const by char string
* String(int);////paramtersied const by int
* String(String &);//copy constructor
* ~String();//Destructor
* void setString(char \*);//method used if we inaitlised string with size

* int GetCapacity();//returns total capacity
* int GetLength();//return length("\0" included)
* char CharAt(int);//returns a char value at the given index number
* void concat(char \*);//combines specified string at the end of this string.

//It returns combined string

* void Display();//Displays string
* void trim();//method eliminates leading and trailing spaces

* int contains(char \*); //conatains function in java

* String\* substring(int, int);

//return all the characters from startIndex to endIndex

* String\* substring(int);//return all the characters from startIndex

* void toLowerCase();//returns the string in lowercase letter
* void toUpperCase();//returns the string in uppercase letter
* void FirstCap();//returns the string in uppercase Fisrt letter of each Word

* void increseCapacity(int);

* BOOL equals(String);//method compares the two given strings based on the content of the string. If any character is not matched, it returns false //If all characters are matched, it returns true

* int compareTo(String);

/\*if s1 > s2, it returns positive number

if s1 < s2, it returns negative number

if S1 == s2, it returns 0

String s1="hello";

String s2="hello";

String s3="meklo";

String s4="hemlo";

String s5="flag"

s1.compareTo(s2));//0 because both are equal

s1.compareTo(s3));//-5 because "h" is 5 times lower than "m"

s1.compareTo(s4));//-1 because "l" is 1 times lower than "m"

s1.compareTo(s5));//2 because "h" is 2 times greater than "f" \*/

* const char\* replace(char old, char newc);//replaces only single character in String
* const char\* replaceFirst(char\* old, char\* newc);//replaces only first instances of char\_String
* const char\* replaceAll(char\* old, char\* newc);//replaces all instances of char\_String

* char\*\* split(char ,int);// method splits this string against given regular expression

//and returns a char array

**string\_man\_logical\_functions:**

* + StringMan(char \*ptr) :String(ptr) {}
  + StringMan(int size) :String(size) {}
  + BOOL isAnagram(char \*);
  + BOOL charAnagram(char \*);
  + BOOL chkPalindrome();