C++ String Class

Dr. Md. Humayun Kabir
CSE Department, BUET

String

- A string is a sequence of characters.
- We use null terminated character arrays (C-strings or C-style strings) to store and manipulate strings.
- We include <cstring> to use C-style string functions, such as strlen(), strcpy etc.
- ANSI C++ provides a class called **string**.
- We must include <string> in our program to create objects from class string

Available Operations on string

- Creating string objects.
- Reading string objects from keyboard.
- Displaying string objects to the screen.
- Finding a substring from a string.
- Modifying string objects.
- Adding string objects.
- Accessing characters in a string.
- Obtaining the size of string.
- And many more.

Commonly Used String Constructors

- String();
  - // For creating an empty string.
- String (const char *str);
  - // For creating a string object from a null-terminated string.
- String(const string &str);
  - // For creating a string object from other string object.

Operator overloaded in string class

Operator	Meaning
=	Assignment
+	Concatenation
+=	Concatenation assignment
==	Equality
!=	Inequality
<	Less than
<=	Less than equal

Operator overloaded in string class

Operator	Meaning
>	Greater than
>=	Greater than equal
[ ]	Subscripting
<<	Output
>>	Input

Creating String Objects

- string s1, s3; // Using constructor with no arguments.
- string s2("xyz"); // Using one-argument constructor.
- s1 = s2; // Assigning string objects
- s3 = "abc" + s2; // Concatenating strings

- cin >> s1; // Reading from keyboard (one word)
- cout << s2; // Display the content of s2
- getline(cin, s1) // Reading from keyboard a line of text

- s3 += s1; // s3 = s3 + s1;
- s3 += "abc"; // s3 = s3 + "abc";

Manipulating String Objects

- string &assign(const string &strob, size_type start, size_type num)
- string &assign(const char *str, size_type num)
- string &append(const string &strob, size_type start, size_type num)
- string &append(const char *str, size_type num)
- string &insert(size_type, start, const string &strob)

Manipulating String Objects

- string &insert(size_type, start, const string &strob, size_type inStart, size_type num)
- string &replace(size_type start, size_type num, const string &strob)
- string &replace(size_type start, size_type num, const string &strob, size_type replaceStrat, size_type replaceNum)
- string &erase(size_type start=0, size_type num=npos) //npos=-1

Manipulating String Objects

- size_type find(const string &strob, size_type start=0) const
- size_type rfind(const string &strob, size_type start=npos) const
- Int compare(size_type start, size_type num, const string &strob) const
- const char *c_str() const
- void swap (string &strob)

Manipulating String Objects

- string s1(“12345”);
- string s2(“abcde”);

- s1.insert(4, s2); // s1 = 1234abcde5

- s1.erase(4, 5); // s1 = 12345

- s2.replace(1, 3, s1); // s2 = a12345e

String Characteristics

size()	Number of elements currently stored
length()	Number of elements currently stored
capacity()	Total elements that can be stored
max_size()	Maximum size of a string object that a system can support
empty()	Return true or 1 if the string is empty otherwise returns false or 0
resize()	Used to resize a string object (effects only size and length)

String Characteristics

```
void display(string &str)
{
    cout << “Size = ” << str.size() << endl;
    cout << “Length = ” << str.length() << endl;
    cout << “Capacity = ” << str.capacity() << endl;
    cout << “Max Size = ” << str.max_size() << endl;
    cout << “Empty: ” << (str.empty() ? “yes” : “no”) << endl;
    cout << endl << endl;
}
```

Accessing Characters in Strings

Function	Task
at()	For accessing individual characters
substr()	For retrieving a substring
find()	For finding a specific substring
find_first_of( )	For finding the location of first occurrence of the specific character(s)
find_last_of( )	For finding the location of last occurrence of the specific character(s)
[] operator	For accessing individual character. Makes the string object to look like an array.

Accessing Characters in Strings

- const char& at (size_t pos) const
- char& at (size_t pos)
- size_t find_first_of (const string& str, size_t pos = 0) const
- size_t find_first_of (const char* s, size_t pos, size_t n) const
- size_t find_first_of (const char* s, size_t pos = 0) const
- size_t find_first_of (char c, size_t pos = 0) const
- string substr (size_t pos = 0, size_t n = npos) const;

cout<<“Thank You”<<endl;

cout<<“**Have a Good Day**”<<endl;