

Strings

Cadenas de caracteres en C++

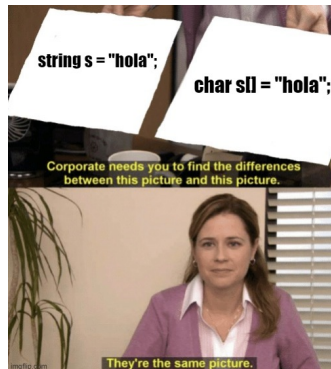
Tomás Peiretti

Strings

Los strings son objetos que representan una secuencia de caracteres, una **cadena de caracteres**. Se los pueden pensar como un arreglo de caracteres, pero que incluyen una gran cantidad de métodos que facilitan las operaciones sobre ellos.

Algunas operaciones:

- **clear()** permite vaciar el string
- **size()** permite saber el tamaño del string
- **copy()** permite copiar un string
- **find()** permite encontrar la posición de un elemento



String: métodos

Las operaciones que ofrecen los **string** son métodos. Los métodos son similares a las funciones, pero se aplican sobre un string en particular:

```
1 int main() {
2     string cadena = "aedd practica strings";
3
4     // imprimir el tamaño de cadena (21)
5     cout << cadena.size() << endl;
6
7     // imprimir la posición donde comienza la cadena "practica" (5)
8     cout << cadena.find("practica") << endl;
9
10    // reemplazar "aedd" por "AEDD"
11    cadena.replace(0,4,"AEDD");
12    cout << cadena << endl; // imprime: AEDD practica strings
13
14    // limpiar la cadena
15    cadena.clear();
16    cout << cadena << endl; // no imprime nada, porque la cadena quedó vacía
17
18    return 0;
19 }
```

String: ejemplos

```
1 void recorrerCadena(string cadena) {
2     // los strings pueden trabajarse como arreglos de caracteres, entonces:
3     for (int i=0; i < cadena.size(); i++) {
4         char caracter = cadena[i];
5         cout<<"cadena[" << i << "] = " << caracter << endl;
6     }
7 }
8 // por defecto, los strings pasan por copia!
9 // si deseamos modificarlo debemos indicar '&'
10 void invertirCadena(string &cadena) {
11     int i=0, j=cadena.size()-1;
12     while(i<j) {
13         swap(cadena[i], cadena[j]);
14         i++; j--;
15     }
16 }
17
18 void ejemploOperadores() {
19     string s1 = "gato", s2 = "perro", s3;
20     // el operador '+' concatena dos strings
21     cout << s1 + s2 + "321" << endl; // imprime: gatoperro321
22     // es posible leer strings por teclado
23     cin >> s3;
24     // y tambien, los strings se pueden comparar con los operadores relacionales
25     if (s3 == s2) {
26         cout << "s3 es igual a s2" << endl;
27     }
28     if (s3 > s2) {
29         cout << "s3 es lexicograficamente mayor que s2" << endl;
30     }
31 }
```

String: ejemplo

```
1 // leer una linea entera y,
2 // si en la linea se encuentra la frase "aedd es lo mejor"
  imprimir: "algo de ritmo y locura de datos"
3 int main() {
4
5     string linea;
6     // la funcion getline nos permite leer la linea entera
7     getline(cin, linea);
8
9     // el metodo find() nos permite saber si se encuentra la
    frase o no
10    // si retorna npos, significa que no lo pudo encontrar
11    if (linea.find("aedd es lo mejor") == string::npos) {
12        cout << "la frase no se encuentra" << endl;
13    }
14    else {
15        cout << "algo de ritmo y locura de datos" << endl;
16    }
17
18 }
```

Métodos de String

Capacity:

size	Return length of string (public member function)
length	Return length of string (public member function)
max_size	Return maximum size of string (public member function)
resize	Resize string (public member function)
capacity	Return size of allocated storage (public member function)
reserve	Request a change in capacity (public member function)
clear	Clear string (public member function)
empty	Test if string is empty (public member function)
shrink_to_fit	Shrink to fit (public member function)

Element access:

operator[]	Get character of string (public member function)
at	Get character in string (public member function)
back	Access last character (public member function)
front	Access first character (public member function)

fix Non-member function overloads

operator+	Concatenate strings (function)
relational operators	Relational operators for string (function)
swap	Exchanges the values of two strings (function)
operator>>	Extract string from stream (function)
operator<<	Insert string into stream (function)
getline	Get line from stream into string (function)

Modifiers:

operator+=	Append to string (public member function)
append	Append to string (public member function)
push_back	Append character to string (public member function)
assign	Assign content to string (public member function)
insert	Insert into string (public member function)
erase	Erase characters from string (public member function)
replace	Replace portion of string (public member function)
swap	Swap string values (public member function)
pop_back	Delete last character (public member function)

String operations:

c_str	Get C string equivalent (public member function)
data	Get string data (public member function)
get_allocator	Get allocator (public member function)
copy	Copy sequence of characters from string (public member function)
find	Find content in string (public member function)
rfind	Find last occurrence of content in string (public member function)
find_first_of	Find character in string (public member function)
find_last_of	Find character in string from the end (public member function)
find_first_not_of	Find absence of character in string (public member function)
find_last_not_of	Find non-matching character in string from the end (public member function)
substr	Generate substring (public member function)
compare	Compare strings (public member function)

<https://cplusplus.com/reference/string/string/>

Ejercicios

- A Contar Letras
- Ola ivan
- Bajando los caracteres
- Invertir Palabra
- Bit de paridad
- Buscando la inicial
- SEQUETAV
- Frecuencia de una letra
- Creo que ya funciona
- Recortando palabras
- Dias de la actualidad
- Cuenta cadenas
- Abreviando palabras
- Encontrando Anagramas
- El caballo de John Carter
- Bailando con Cesar