

Repaso de Funciones de Cadenas en C++

1. `std::string::length` o `std::string::size`

Esta función devuelve el número de caracteres en la cadena.

Ejemplo 1

```
std::string str = "Hola Mundo";
std::cout << "La longitud de la cadena es: " << str.length() << std::endl;
```

Ejemplo 2

```
std::string str = "C++ es genial";
std::cout << "El tamaño de la cadena es: " << str.size() << std::endl;
```

Ejemplo 3

```
std::string str = "";
std::cout << "El tamaño de la cadena vacía es: " << str.size() << std::endl;
```

2. `std::string::substr`

Esta función devuelve una subcadena de la cadena original, comenzando en un índice y con una longitud específica.

Ejemplo 1

```
std::string str = "Hola Mundo";
std::string sub = str.substr(0, 4); // "Hola"
std::cout << "La subcadena es: " << sub << std::endl;
```

Ejemplo 2

```
std::string str = "C++ es genial";
std::string sub = str.substr(4, 2); // "es"
std::cout << "La subcadena es: " << sub << std::endl;
```

Ejemplo 3

```
std::string str = "Aprendiendo C++";
std::string sub = str.substr(11); // "C++"
std::cout << "La subcadena es: " << sub << std::endl;
```

3. `std::string::find`

Esta función busca la primera aparición de una subcadena o carácter en la cadena y devuelve su posición.

Ejemplo 1

```
std::string str = "Hola Mundo";
size_t found = str.find("Mundo");
if (found != std::string::npos)
    std::cout << "Se encontró 'Mundo' en la posición: " << found << std::endl;
```

Ejemplo 2

```
std::string str = "C++ es genial";
size_t found = str.find("es");
if (found != std::string::npos)
    std::cout << "Se encontró 'es' en la posición: " << found << std::endl;
```

Ejemplo 3

```
std::string str = "Aprendiendo C++";
size_t found = str.find("Java");
if (found == std::string::npos)
    std::cout << "'Java' no se encontró en la cadena." << std::endl;
```

4. std::string::replace

Esta función reemplaza una parte de la cadena con otra cadena.

Ejemplo 1

```
std::string str = "Hola Mundo";
str.replace(5, 5, "Amigo");
std::cout << "Cadena después de reemplazo: " << str << std::endl;
```

Ejemplo 2

```
std::string str = "C++ es genial";
str.replace(0, 3, "Java");
std::cout << "Cadena después de reemplazo: " << str << std::endl;
```

Ejemplo 3

```
std::string str = "Aprendiendo C++";
str.replace(11, 3, "Python");
std::cout << "Cadena después de reemplazo: " << str << std::endl;
```

5. std::string::append

Esta función añade una cadena al final de la cadena original.

Ejemplo 1

```
std::string str = "Hola";
str.append(" Mundo");
std::cout << "Cadena después de append: " << str << std::endl;
```

Ejemplo 2

```
std::string str = "C++";
str.append(" es genial");
std::cout << "Cadena después de append: " << str << std::endl;
```

Ejemplo 3

```
std::string str = "Aprendiendo";
str.append(" C++");
std::cout << "Cadena después de append: " << str << std::endl;
```

6. std::string::compare

Esta función compara dos cadenas lexicográficamente.

Ejemplo 1

```
std::string str1 = "Hola";
std::string str2 = "Hola";
if (str1.compare(str2) == 0)
    std::cout << "Las cadenas son iguales" << std::endl;
```

Ejemplo 2

```
std::string str1 = "Hola";
std::string str2 = "Adiós";
if (str1.compare(str2) > 0)
    std::cout << "La cadena '" << str1 << "' es mayor que '" << str2 << "'" <<
    std::endl;
```

Ejemplo 3

```
std::string str1 = "Hola";
std::string str2 = "Mundo";
if (str1.compare(str2) < 0)
    std::cout << "La cadena '" << str1 << "' es menor que '" << str2 << "'" <<
    std::endl;
```

7. std::string::erase

Esta función elimina una parte de la cadena, especificada por un índice y una longitud.

Ejemplo 1

```
std::string str = "Hola Mundo";
str.erase(5, 5);
std::cout << "Cadena después de erase: " << str << std::endl;
```

Salida esperada: Cadena después de erase: Hola

Ejemplo 2

```
std::string str = "C++ es genial";
str.erase(4, 2);
std::cout << "Cadena después de erase: " << str << std::endl;
```

Salida esperada: Cadena después de erase: C++ genial

Ejemplo 3

```
std::string str = "Aprendiendo C++";
str.erase(0, 11);
std::cout << "Cadena después de erase: " << str << std::endl;
```

Salida esperada: Cadena después de erase: C++

8. std::string::insert

Esta función inserta una subcadena en una posición específica de la cadena.

Ejemplo 1

```
std::string str = "Hola Mundo";
str.insert(5, "Amigo ");
std::cout << "Cadena después de insert: " << str << std::endl;
```

Salida esperada: Cadena después de insert: Hola Amigo Mundo

Ejemplo 2

```
std::string str = "C++ es genial";
str.insert(3, " lenguaje");
std::cout << "Cadena después de insert: " << str << std::endl;
```

Salida esperada: Cadena después de insert: C++ lenguaje es genial

Ejemplo 3

```
std::string str = "Aprendiendo";
str.insert(11, " C++");
std::cout << "Cadena después de insert: " << str << std::endl;
```

Salida esperada: Cadena después de insert: Aprendiendo C++

9. std::string::toupper (Usando std::transform)

Convierte todos los caracteres de la cadena a mayúsculas.

Ejemplo 1

```
std::string str = "Hola Mundo";
std::transform(str.begin(), str.end(), str.begin(), ::toupper);
std::cout << "Cadena en mayúsculas: " << str << std::endl;
```

Salida esperada: Cadena en mayúsculas: HOLA MUNDO

Ejemplo 2

```
std::string str = "C++ es genial";
std::transform(str.begin(), str.end(), str.begin(), ::toupper);
std::cout << "Cadena en mayúsculas: " << str << std::endl;
```

Salida esperada: Cadena en mayúsculas: C++ ES GENIAL

Ejemplo 3

```
std::string str = "Aprendiendo C++";
std::transform(str.begin(), str.end(), str.begin(), ::toupper);
std::cout << "Cadena en mayúsculas: " << str << std::endl;
```

Salida esperada: Cadena en mayúsculas: APRENDIENDO C++

10. std::string::tolower (Usando std::transform)

Convierte todos los caracteres de la cadena a minúsculas.

Ejemplo 1

```
std::string str = "Hola Mundo";
std::transform(str.begin(), str.end(), str.begin(), ::tolower);
std::cout << "Cadena en minúsculas: " << str << std::endl;
```

Salida esperada: Cadena en minúsculas: hola mundo

Ejemplo 2

```
std::string str = "C++ es genial";
std::transform(str.begin(), str.end(), str.begin(), ::tolower);
std::cout << "Cadena en minúsculas: " << str << std::endl;
```

Salida esperada: Cadena en minúsculas: c++ es genial

Ejemplo 3

```
std::string str = "Aprendiendo C++";
std::transform(str.begin(), str.end(), str.begin(), ::tolower);
std::cout << "Cadena en minúsculas: " << str << std::endl;
```

Salida esperada: Cadena en minúsculas: aprendiendo c++

11. `std::string::at`

Esta función devuelve una referencia al carácter en la posición especificada.

Ejemplo 1

```
std::string str = "Hola Mundo";  
char ch = str.at(1);  
std::cout << "Carácter en la posición 1: " << ch << std::endl;
```

Salida esperada: Carácter en la posición 1: o

Ejemplo 2

```
std::string str = "C++ es genial";  
char ch = str.at(4);  
std::cout << "Carácter en la posición 4: " << ch << std::endl;
```

Salida esperada: Carácter en la posición 4: e

Ejemplo 3

```
std::string str = "Aprendiendo C++";  
char ch = str.at(11);  
std::cout << "Carácter en la posición 11: " << ch << std::endl;
```

Salida esperada: Carácter en la posición 11: C