Strings

C++ provides support for strings using two main approaches: C-style strings and the `std::string` class from the Standard Library. The `std::string` class is preferred for most applications due to its safety and flexibility.

1 C-style Strings

A C-style string is an array of characters terminated by the null character \`0`. It is declared using a character array.

```cpp
char greeting[] = "Hello";
```

`cstring` library functions like `strlen()`, `strcpy()`, and `strcmp()` are used to manipulate C-style strings.

2 std::string Class

The `std::string` class is part of the `<string>` header and provides a wide range of operations.

- Safe and dynamic resizing.
- Supports operators like `+` for concatenation and `==` for comparison.
- Can use member functions like `length()`, `substr()`, `find()`, and `append()`.

```cpp
#include <iostream>
#include <string>
using namespace std;

int main() {
    string name = "Alice";
    string greeting = "Hello, " + name;
    cout << greeting << endl;
    return 0;
}
```

3 Common Operations

```cpp
string s = "example";
int len = s.length();        // Get length
char c = s[0];               // Access character
s += " string";              // Append
s.substr(2, 3);              // "amp"
s.find("amp");               // Returns 2
```