rules.sonarsource.com

C++ static code analysis: "starts_with" and "ends_with" should be used for prefix and postfix checks

2 minutes

In C++20 `std::string` and `std::string_view` gain new member functions `starts_with` and `ends_with` that compare their argument to the prefix and postfix of the string.

These two functions introduce a standard, concise, and efficient way of checking the prefix and postfix for strings. The ad-hoc implementations predating C++20 often are often less readable, less efficient, and less reliable.

This rule reports places that check prefix or postfix of a string using an ad-hoc implementation.

Noncompliant Code Example

```
if (6 <= str.size() && str.substr(0, 6) == "prefix") { // Noncompliant
  std::cout <<str <<" starts with the prefix\n";
}
if (6 <= str.size() && std::string_view(str.begin(), str.begin() + 6) ==
"prefix") { // Noncompliant
  std::cout <<str <<" starts with the prefix\n";
}
if (7 <= str.size() && str.substr(str.size() - 7) == "postfix") { //
Noncompliant
  std::cout <<str <<" ends with the postfix\n";
}
```

Compliant Solution

```
if (str.starts_with("prefix")) {
  std::cout <<str <<" starts with the prefix\n";
}
if (str.ends_with("postfix")) {
  std::cout <<str <<" ends with the postfix\n";
}
```