



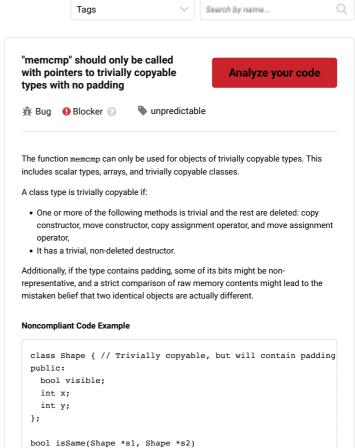
C++ static code analysis

Unique rules to find Bugs, Vulnerabilities, Security Hotspots, and Code Smells in your C++ code





"pthread_mutex_t" should be properly



Compliant Solution

```
class Shape {
public:
   bool visible;
   int x;
   int y;
};

bool operator==(Shape const &s1, Shape const &s2) {
   return s1.visible == s2.visible && s1.x == s2.x && s1.y == :
}

bool isSame(Shape *s1, Shape *s2) {
   return (*s1) == (*s2);
}
```

return memcmp(s1, s2, sizeof Shape) == 0; // Noncompliant

Available In:

sonarlint ⊖ sonarcloud ♦ sonarqube Develor Edition

initialized and destroyed



"pthread_mutex_t" should not be consecutively locked or unlocked twice



"std::move" and "std::forward" should not be confused

🕦 Bug

A call to "wait()" on a "std::condition_variable" should have a © 2008-2022 SonarSource S.A., Switzerland. All content is copyright protected. SONAR, SONARSOURCE, SONARLINT, SONARQUBE and SONARCLOUD are trademarks of SonarSource S.A. All other trademarks and copyrights are the property of their respective owners. All rights are expressly reserved.

Privacy Policy