

Qt 创建者手册 > [使用 Cpp 检查分析代码](#)

使用 Cpp 检查分析代码

Cppcheck 是一种静态分析工具，用于检测 C++ 代码中的错误。静态分析是在源代码上执行的，而不实际执行应用程序。

实验性 Cppcheck 诊断插件将 Cppcheck 工具生成的诊断集成到 C++ 编辑器中。

Cppcheck 自动对打开的文件运行。若要选择要签入当前活动项目的文件，请选择“**分析** > Cppcheck”。

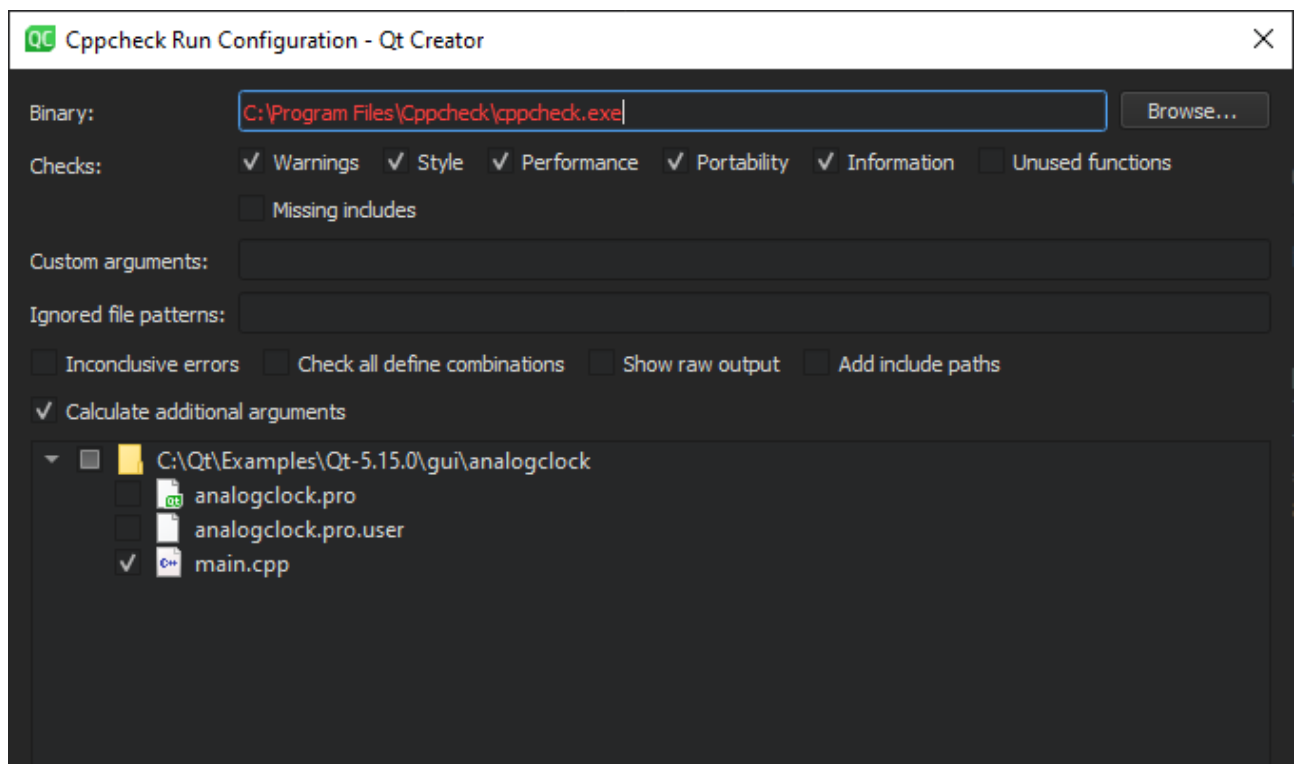
启用 Cpp 检查插件

要启用 Cppcheck 插件：

1. 选择“**帮助** > **关于插件**” > **代码分析器** > Cppcheck 以启用插件。
2. 选择**立即重新启动**以重新启动 Qt 创建器并加载插件。

对所选文件运行 Cpp 检查

1. 选择“**分析** > **Cpp 检查**”。



- 在“**二进制**”字段中，输入 Cppcheck 可执行文件的路径。
- 在“**检查**”组中，选择要执行的检查。

注意：默认情况下，Cppcheck 使用多个线程来执行检查。选择“**未使用的函数**”选项将禁用默认行为。

- 在“**自定义参数**”字段中，输入用于运行 Cppcheck 的其他参数。这些参数可能会被自动生成的参数所掩盖。若要避免配置中可能发生的冲突，请选中“**显示原始输出**”复选框以查看最终参数。
- 在“**忽略的文件模式**”字段中，输入用于忽略与模式匹配的文件（通配符）的筛选器。您可以输入多个模式，以逗号分隔。即使 Cppcheck 未在与提供的模式匹配的文件上运行，如果其他文件包含它们，则可能会隐式检查它们。
- 选中“**不确定错误**”复选框以标记可能的误报。
- 选中“**检查所有定义组合**”复选框以选中“全部定义组合”。启用此选项可能会显著减慢分析速度，但可能有助于发现更多问题。
- 选中“**添加包含路径**”复选框以将当前项目的包含路径传递给 Cppcheck。启用此选项会减慢对大型项目的检查速度，但可以帮助 Cppcheck 查找缺少的包含项。
- 选中“**计算其他参数**”复选框，根据当前项目的设置（如使用的语言 and 标准版本）计算其他参数，并将其传递给 Cppcheck。
- 选择要对其运行 Cppcheck 的文件。
- 选择“**分析**”。

Qt Creator对所选文件运行Cppcheck，并通过文本标记或注释显示结果。

若要为自动运行检查指定上述设置，请选择“**编辑>首选项>分析器>Cppcheck**”。

[< 分析中央处理器使用率](#)

[可视化浏览器跟踪事件 >](#)

©2022 Qt Ltd. 此处包含的文档贡献是其各自所有者的版权。此处提供的文档是根据自由软件基金会发布的 [GNU 自由文档许可证 1.3 版](#) 的条款进行许可的。Qt及其相应的徽标是Qt有限公司在芬兰和/或全球其他国家的商标。所有其他商标均为其各自所有者的财产。



Contact Us

Company

[About Us](#)

[Investors](#)

[Newsroom](#)

Licensing

[Terms & Conditions](#)

[Open Source](#)

[FAQ](#)



Support

- Support Services
- Professional Services
- Partners
- Training

For Customers

- Support Center
- Downloads
- Qt Login
- Contact Us
- Customer Success

Community

- Contribute to Qt
- Forum
- Wiki
- Downloads
- Marketplace