

# SLR(1)分析程序编译指南

## SLR(1)分析程序编译指南

- 1. 编译环境
- 2. 编译步骤
- 3. 故障排除

## 1. 编译环境

- 1. 操作系统：macOS Sonoma 14.0+ 或 Windows 10+
- 2. 依赖套件：
  - Qt 6.6
  - Xcode-Command-Line-Tools (macOS)

## 2. 编译步骤

- 1. 在Qt Creator中打开项目（LR\_SLR.pro），选择目标环境；
- 2. 选择构建Release版本
- 3. 在项目代码目录的上一级目录可以看到输出文件，进入带“Release”后缀的目录。











- 4. 【macOS】可以看到输出的可执行文件，在macOS下，是 .app 后缀，双击即可运行



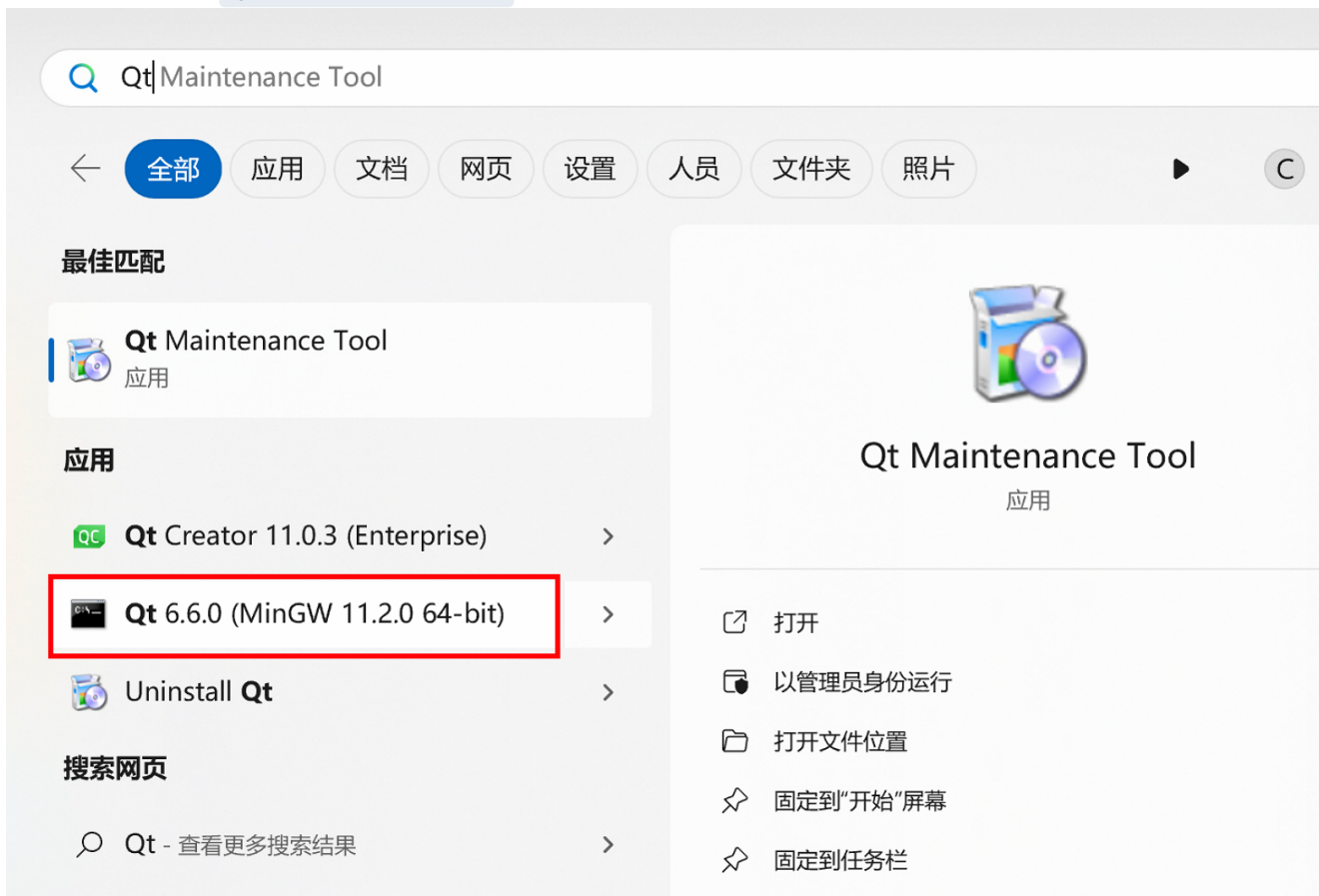
如果是Windows系统，还需要进行一下步骤：

5. 【Windows】进入Release目录

名称	修改日期	类型
 .qtc_clangd	2023/11/22 18:04	文件夹
 debug	2023/11/22 18:04	文件夹
 <b>release</b>	2023/11/22 18:04	文件夹
 .qmake.stash	2023/11/22 18:04	STASH 文件
 Makefile	2023/11/22 18:04	文件
 Makefile.Debug	2023/11/22 18:04	DEBUG 文件
 Makefile.Release	2023/11/22 18:04	RELEASE 文件
 ui_dialog.h	2023/11/22 18:04	QtProject.QtCre..

可以看到 `.exe` 文件，此时还需要进行下一步部署。

6. 【Windows】打开 Qt 6.6.0 (MinGW64) 命令行工具（在安装Qt时会集成）



7. 【Windows】进入编译输出的这个目录，执行 `windeployqt LR_SLR.exe` 部署  
等待部署完成即可，回到编译输出的目录，双击 `.exe` 文件即可打开。

### 3. 故障排除

1. Windows下在构建结束后必须按照上面的步骤在Qt命令行执行 `windeployqt` 进行部署，并且产生的 `dll` 文件不可删除；
2. macOS下需要安装Xcode-Command-Line-Tools