

VSCode 小帮手

什么是BUG,能运行的就是队 QT大作业 作业报告

组长：顾嘉琪

组员：秦慕予，孙煜琦

快速转至对应位置：

- [项目各模块与类设计细节](#)
- [小组成员分工情况](#)
- [项目总结与反思](#)

程序功能介绍

VSCode 小帮手程序提供了快捷配置VS Code环境以及一些其它辅助功能。

截至大作业提交时，实现的功能有：

- VS Code 的C++单文件编译和调试配置，并提供了一些自定义选项
- VS Code 的Python使用准备，Python相关插件推荐以及Python下载指引
- VS Code 的基于conda的Python多版本管理帮助
- VS Code 的copilot辅助编程和对话教程
- VS Code 的离线C++和基于conda的Python多版本（3.6+）配置支持

以及为了充分运用程序已有能力提供的越过VS Code前置要求的Python安装辅助和conda安装及环境激活辅助

功能设计上，我们力图实现“对于计算机基础完全不了解的人，也能在指引下一步步完成VS Code的配置”，提供了压缩文件格式.7z的解压说明和7-Zip下载连接，C++工作文件夹路径非ASCII字符检测等功能。对于最极端情况，一键通功能使得使用者不需要考虑一切过程，由程序代为完成相应诸多工作。同时，我们也希望对相应需求略有了解的同学能得到相应定义自己的VS Code的空间，因而我们也提供了部分自定义选项，对于非必要性的插件和操作提供了可选空间。

需要承认的是，程序开发时间有限，目前仍然存在不足之处，如Python的相关指引存在缺失，可能使用者并不了解如何进行操作，也没有进行相应的快捷键绑定等；再比如程序没能提供对于conda环境的

移除功能，用户若激活了一个不想要的环境，目前还只能上网了解相关知识后自行移除。

项目各模块与类设计细节

本项目源代码中所含文件作用如下：

head files:

json_operator.h

内部定义了Json_operator类，用于提供处理VS Code所用的json文件的函数

path_operator.h

内部定义了Path_operator类，用于提供一个处理路径相关问题的函数

string_operator.h

内部定义了String_operator类，用于提供字符串处理所用的函数

mainwindow.h

定义本程序所用的MainWindow类

source files:

main.cpp

程序入口

mainwindow.cpp

内含程序中所有的槽函数和处理相应问题的一些函数

other files:

CMakeLists.txt

Qt自动生成的用于C++多文件编译的CMake文件

resource.qrc

管理程序所用资源（图片和所用文件）的资源文件

mainwindow.ui

记录程序UI设计的文件，本程序是基于UI以可视化方式制作的图形界面

jsondoc/...

内部含有程序所用的json文件以及一个hello world文件(由.qrc管理)

image/...

内部含有程序所用的图像文件(由.qrc管理)

* offline-install-package.tar.gz

离线资源包文件，不存在于源代码仓库中，随支持离线版本程序分发

本项目所用的类设计基本用于保证项目开发的可并行性，即将可以从槽函数中抽离出的功能单独开发，与槽函数所在的mainwindow.cpp分离交由不同同学同时段修改。但由于开发前期队员没有立即参与到开发中，部分可以分入operator系列类的内容也被放在了mainwindow.cpp内实现，如检测文件存在性、从环境变量中获取路径等操作。项目中每个operator类内部均仅含有一些public的函数，通过先声明出对象的方式直接使用对应函数。

由于本项目没有什么需要存储的数据，大多操作都是直接进行文件操作；因此，本项目没有用于存储数据的类结构，几个需要存储的变量均直接定义在了MainWindow类中直接使用。

小组成员分工情况

- 顾嘉琪(组长)： ui排版设计，文件操作和命令行操作实现，填充其余槽函数内容， ui样式表填充，测试
- 秦慕予： 字符串操作实现，填充其余槽函数内容， ui样式表填充，测试
- 孙煜琦： 进度图片制作

项目总结与反思

通过本项目开发，我们大致了解了通过github进行多人开发的方式，相较于大一上通过微信交流进行黑白棋应用的开发而言，利用git工具和github进行多人合作开发明显变得更加便捷。

本项目在创立之初设计的目标最终基本达成，实现了一个可用的VS Code配置工具，但距离完美还有一段距离。程序目前仍存在部分设计目标尚未达成。考虑到该程序最终目的是最终能作为一个可以为其它同学提供帮助的实用工具，该程序后续仍然需要进一步开发。

开发过程中遇到的主要问题在于Qt中的QProcess对于通过cmd启动指定路径的应用时存在的一定问题，最终仍未能完全解决，而是通过设定环境变量的方式将问题绕开。但通过本项目，我们也切实积累了一些程序间调用的经验。

关于未来开发，本项目主要需要面对的问题是日益增长的mainwindow.cpp文件规模带来的可维护性的下降。由于程序利用了大量的按钮，对应槽函数都直接定义在了该文件中，使得该文件在后续修补BUG时定位难度日趋增加。事实上，Qt中有将槽函数进行绑定的操作，如果能够在开发之初考虑到按钮数量的骤增，可以考虑例如将所有返回按钮绑定到同一函数、将部分与ui交互不紧密的按钮的函数整理后剥离到其它文件等方式。不过对于当前规模的程序，槽函数的修改尚且不成为一个大问题。

该项目开发过程中对于Qt部件的规范命名略有缺失。开发之初部分认为不重要、不需要另行命名的按钮的易混名称事实上成为了后续一个BUG诞生的部分原因。经过对本次项目的反思得到，对于组件的重命名应当进行的更细致、全面，应当避免任何被代码使用的组件使用默认名称的情况。

本项目由于项目本身功能性的原因，在测试上天然存在难度，因而有很多BUG是在开发后测试环节被发现，其中就有认定为基本完善的版本最后实际使用时被找出BUG的v0.3-alpha版本，导致了路演前v0.3.1-alpha版本的紧急更新。这也提醒了我们团队测试的必要性，在后续加强了测试的力度，并在路演上发出了开源的声明、程序的获取方式及测试邀请。

~~（不过事实上，由于人手有限且程序暂未有推广环境，进行的测试仍十分有限）~~

Last but not least, 感谢老师&助教这一学期的付出, 望您能给个高分

Qt大作业组 **什么是BUG, 能运行的就是队** 队长 顾嘉琪 队员 秦慕予 孙煜琦

2024.6.30