个人实践作业1：代码管理实验报告

实验过程：

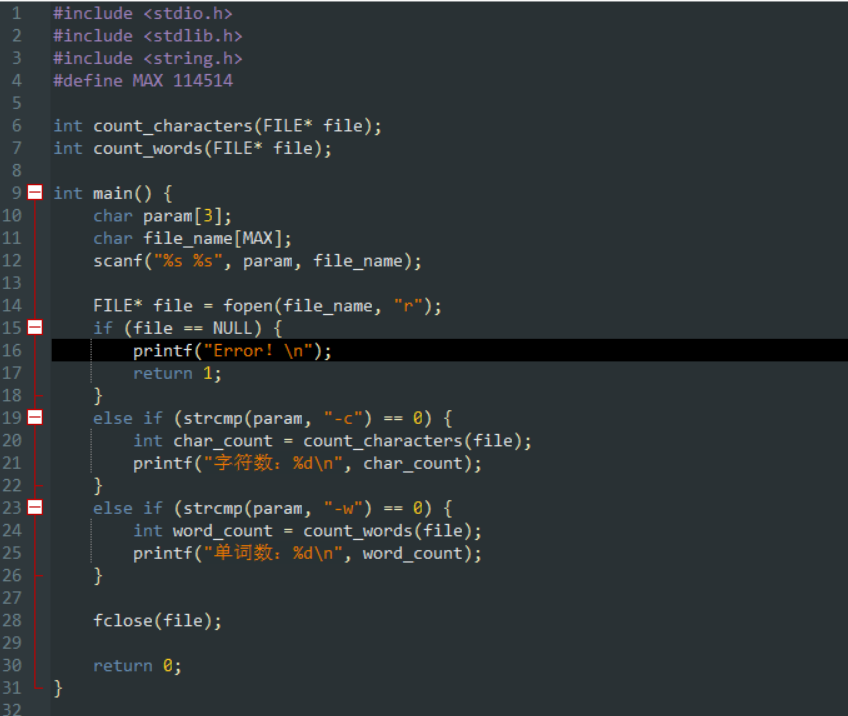
创建github仓库，并安装Git工具：

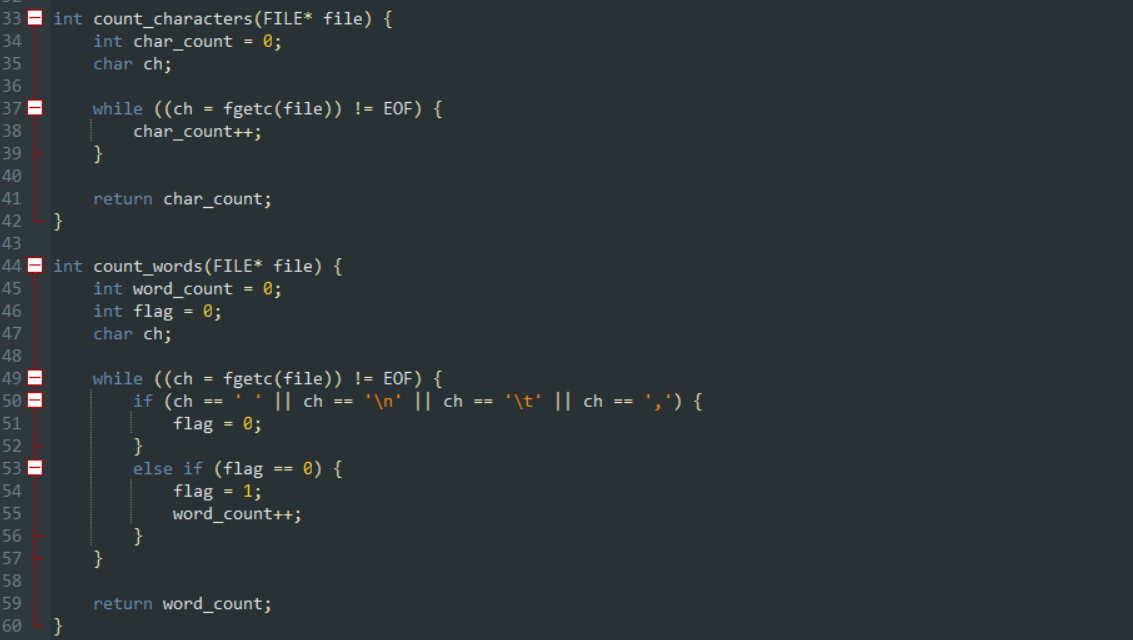
此前已安装过git工具。

Github仓库地址：<https://github.com/wxzhishang/worldCount.git>

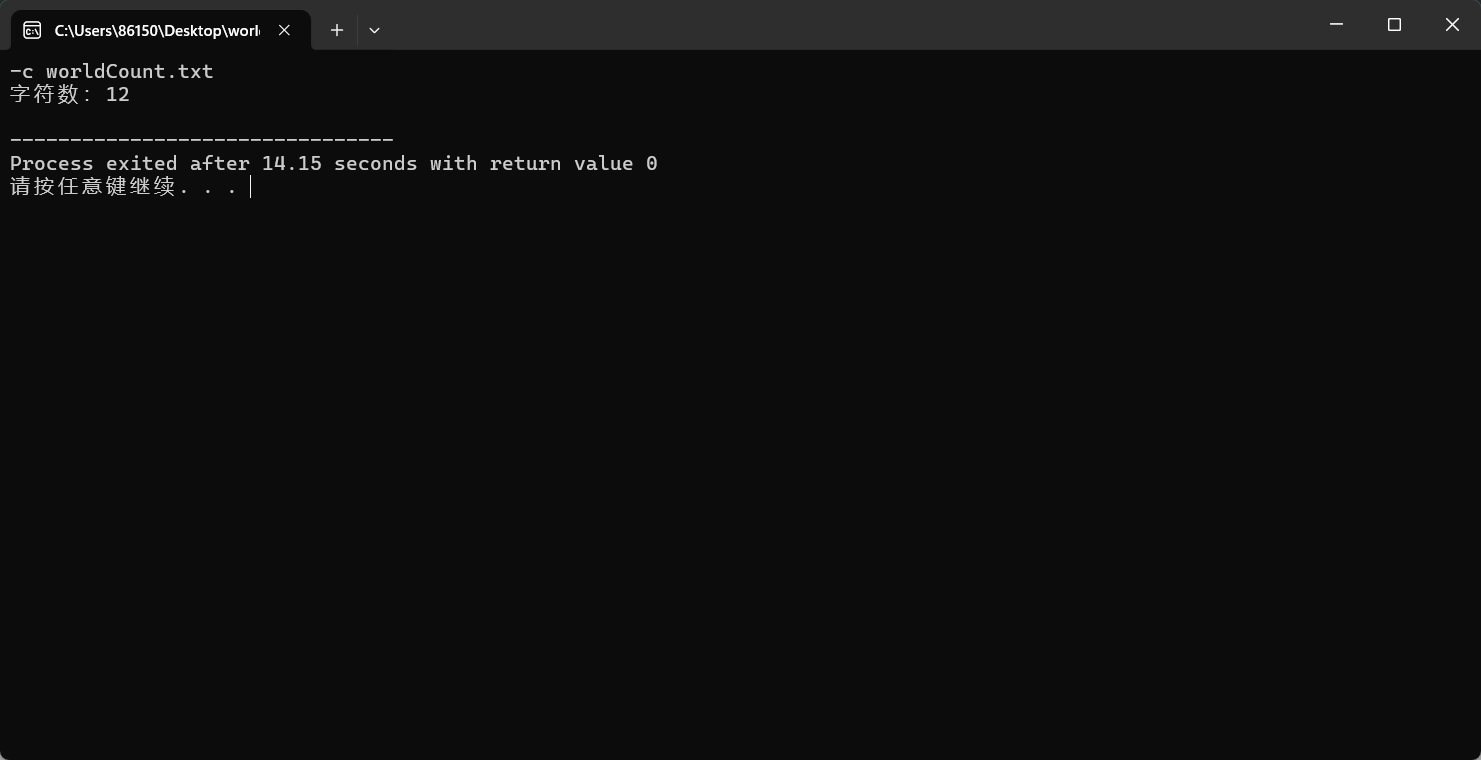
编写代码，并运行结果：

核心代码截图：





运行截图：



Git提交过程：

1. git init：初始化本地仓库；
2. git remote add origin <url>：关联远程仓库；
3. git add .：将所有修改的文件添加到暂存区；
4. git commit -m “code change”：将修改提交到本地仓库，命名本次提交为code change；
5. git push origin master:main：将修改从本地master分支推送到远程仓库的main分支；

本次学习感悟：

近日，我深入学习了Git这一版本控制系统，并进行了实际的代码推送操作。这次实验不仅加深了我对Git工作原理的理解，也让我体会到了团队协作中版本控制的重要性。

一、初识Git

在开始实验之前，我对Git的了解仅限于它是一个流行的版本控制系统。然而，当我真正开始学习和使用时，我才意识到Git的强大和灵活性。它不仅可以记录文件的每一次修改，还可以跟踪这些修改之间的关联，形成一个完整的项目历史记录。这种能力对于团队协作、代码审查以及错误追踪至关重要。

二、实际操作中的挑战

在实际操作中，我遇到了不少挑战。首先，Git的命令行界面对于初学者来说有些复杂，需要花费一定的时间去熟悉和掌握。其次，在推送代码时，我经常遇到各种冲突和错误，如分支冲突、权限问题以及代码风格不一致等。这些问题让我深刻体会到了团队协作中沟通和协调的重要性。

三、解决问题与成长

面对这些挑战，我并没有气馁，而是积极寻求解决方案。我通过查阅官方文档、参加线上课程以及向同事请教等方式，逐渐掌握了Git的基本操作，并学会了如何解决常见的冲突和错误。在这个过程中，我不仅提升了自己的技术能力，还学会了如何与团队成员进行有效的沟通和协作。

四、实验感悟

通过这次实验，我深刻体会到了Git在团队协作中的重要作用。它不仅可以确保代码的一致性和可追踪性，还可以提高团队协作的效率和质量。同时，我也认识到了持续学习和自我提升的重要性。在未来的工作中，我将继续深入学习Git和其他相关技术，以便更好地应对各种挑战和机遇。

此外，我还意识到了团队协作中沟通和协调的重要性。在推送代码时，我们需要与团队成员保持密切的沟通和协作，确保代码的质量和一致性。同时，我们还需要尊重他人的工作和意见，共同为项目的成功而努力。

总之，这次Git推送代码的实验让我受益匪浅。我不仅掌握了Git的基本操作，还学会了如何在团队协作中运用Git进行版本控制。这些经验和感悟将对我未来的工作和学习产生积极的影响。