实验 3：IPC 与 Map-Reduce

截止日期

请参考实验室作业要求。

目标

本项目的目标是练习各种 IPC 方法（用于数据传递和同步）并学习 Map-Reduce（并行计算）。这两者都是工业中经常使用的非常重要的技术。

详情

该项目由三个独立的子项目组成，每个子项目都执行相同的任务：给定一个文本文件，程序输出包含给定单词的行。例如，给定一行“Hello World!”，假设感兴趣的单词是“world”，那么这一行应该被输出。（请注意，如果一行是“Hello worlds”，那么这一行不应该被输出）。
你的程序执行（即父进程）将创建一个子进程。父进程可以打开文本文件，读取内容，并使用以下方法之一将其传递给子进程，但不能检查单词，而子进程不能打开该文件，但可以检查单词。最后，父进程（而不是子进程）应该按字母顺序输出行。
以下是三个子项目的要求：

1. 使用管道作为传递文件内容和结果的方法。
2. 使用 Unix 域套接字作为传递文件内容和结果的方法。
3. 使用共享内存作为传递文件内容和结果的方法。此外，子进程创建 4 个线程，每个线程都充当映射器；而子进程的主线程充当单个归约器。仅此子项目需要 Map-Reduce。不允许使用 Hadoop 作为 Map-Reduce 基础设施；相反，您必须使用 Posix 线程编程来实现 Map-Reduce。

示例输入文本文件

《安娜·卡列尼娜》。列奥·托尔斯泰，1870 年。（见附件：《安娜·卡列尼娜.txt》）一个 650 万字的文件。（见附件 big.txt）
这些只是一些示例文本文件。您的程序应该接受一个文件路径作为参数。换句话说，它应该能够处理任何文本文件。因此，为输入文件创建索引不是个好主意，也是不允许的。

提交

您的提交内容应包括：（1）代码（需要一份 Makefile），（2）一份描述您的设计以及如何编译/使用您的代码的自述文件，以及（3）关于以下家庭作业的报告：

对您的三个程序执行计时，并分析是什么导致了性能上的差异。

☒ 您对该程序的设计
实验结果（统计数据）的截图及分析
☒ 遇到的问题及您的解决方案
总结 Linux 提供的不同进程间通信（IPC）方法，并描述何时使用哪种方法。

写一个关于 Map-Reduce 的简短段落。
写一段关于Hadoop的短文，以及你对它为什么在工业中流行和重要的理解。

☒ 参考资料
您的建议和意见

环境

Linux（推荐使用 Ubuntu 18.04/16.04）和 C/C++ 。

参考文献

你可能会发现以下文章有用。

一个简单的 makefile 教程。科尔比大学。链接
《Linux IPC 入门》。克尔里斯克，2013 年。链接
《Beej 的 Unix 进程间通信指南》。Beej，2010 年。链接
《Linux 进程间通信》。戈尔特、米尔、伯克特和威尔士，1995 年。链接 《MapReduce：大规模集群上简化数据处理》。迪恩和格玛沃特，2004 年。链接
POSIX线程编程。巴尼,2015。链接
《Beej 的 GDB 快速指南》。Beej，2009 年。链接