文件压缩与关键字检索问题实验设计报告

1. 问题描述

功能要求：

给定一个txt文件需要实现以下功能：

1、压缩功能：使用哈夫曼编码对文档中出现的所有英文字符和符号进行编码，产生码表，之后利用码表可以txt格式的英文文档进行压缩编码，输出编码后的二进制文件并给出文件的压缩率。

2、解压功能：对编码后的文件利用码表进行解码。

3、关键词查找功能：对输入的单词在压缩后的二进制文件中查找，如果找到，则给出该英文单词出现的次数和所有位置，如果没有找到返回检索失败。要求使用BM算法

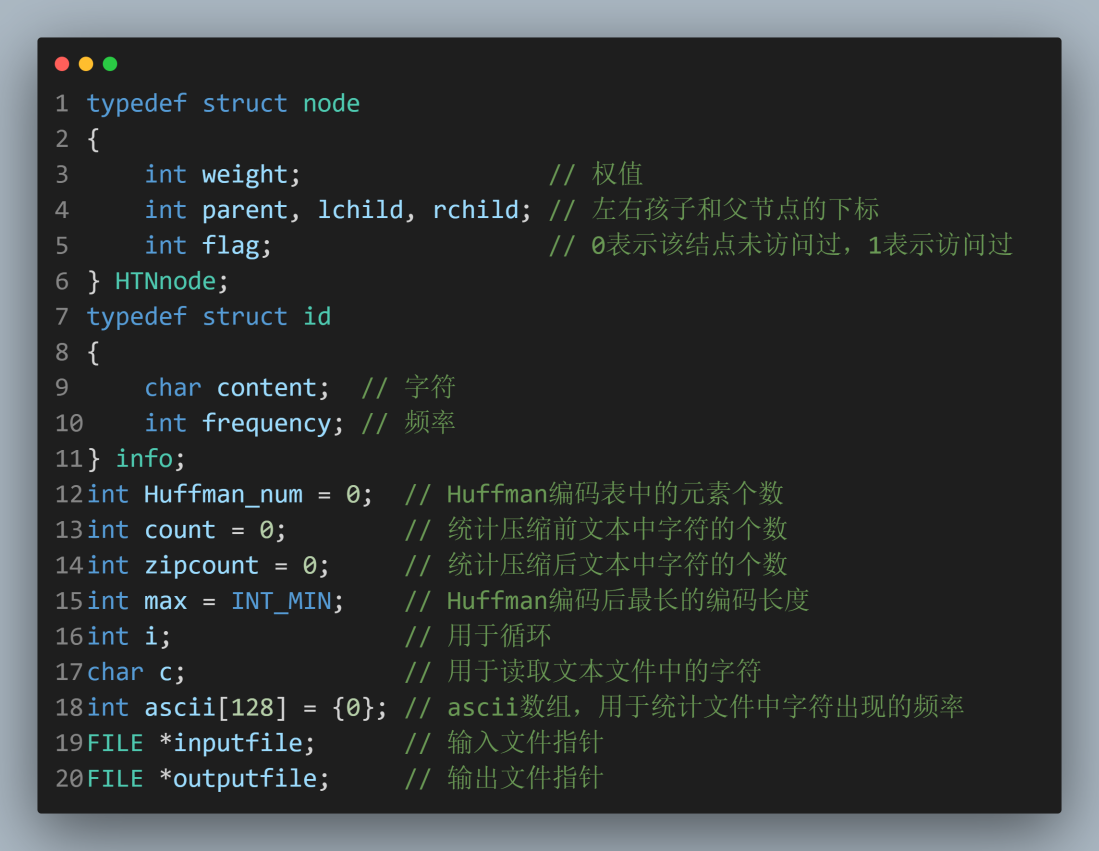
输入要求：

从命令行依次输入英文文档路径和输出压缩文件路径，后输入要查找的单词

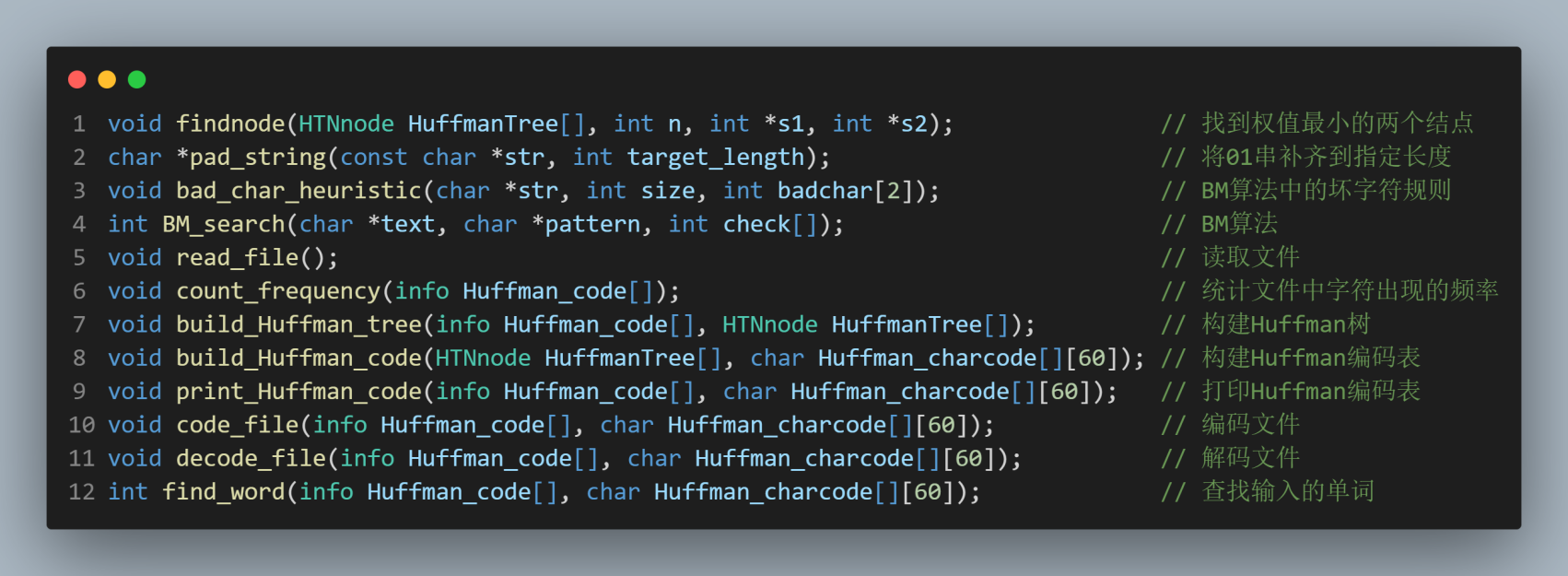
输出要求：

先在指定输出路径输出压缩后的文件，之后在命令行输出从压缩后的文件解码出的内容，之后输入查找结果

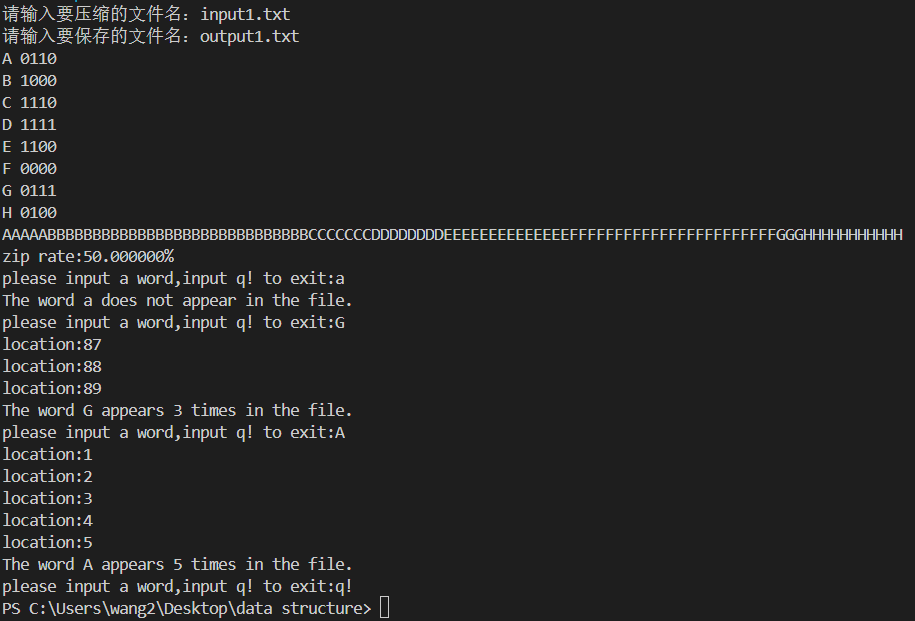
1. 设计思路
2. 读取文件：先存入输入文件和输出文件，创造输入文件指针和输出文件指针
3. 统计文件中出现的字符：使用长度为128的一维数组来统计文件中出现的字符和种类数
4. 统计文件中每种出现的字符的频率：创建结构体用来储存出现的字符及其频率
5. 构建赫夫曼树：按照出现的字符与频率构造赫夫曼树
6. 构建赫夫曼编码表：按照构建出的赫夫曼树构建赫夫曼编码表并打印
7. 编码文件：按照赫夫曼编码表将源文件一个字符一个字符地转化成二进制编码，并在二进制文件中输出
8. 解码文件：按照赫夫曼编码表将二进制文件解码成字符文件，并计算压缩率
9. 查找单词：将输入的单词先按照赫夫曼编码表转换成二进制串，之后使用BM算法在压缩后的文件中查找
10. 代码说明
11. 结构体与全局变量的定义



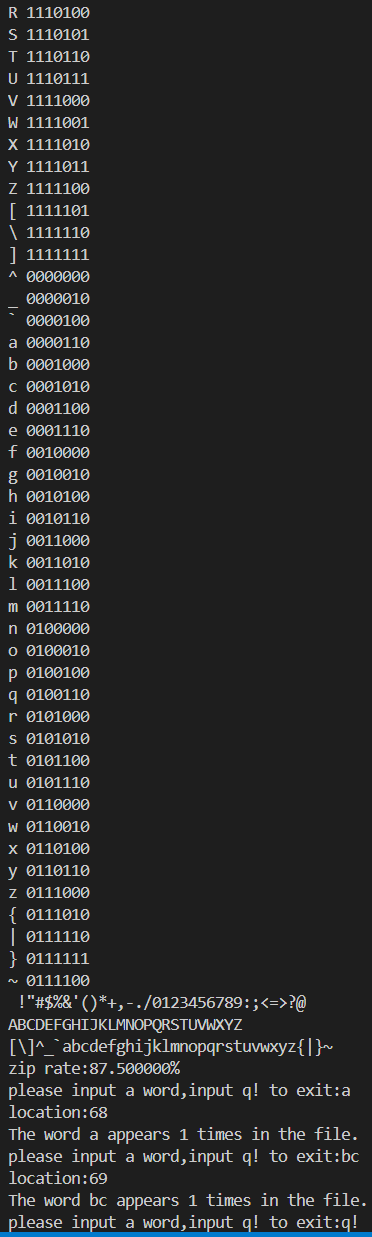
1. 函数定义和功能

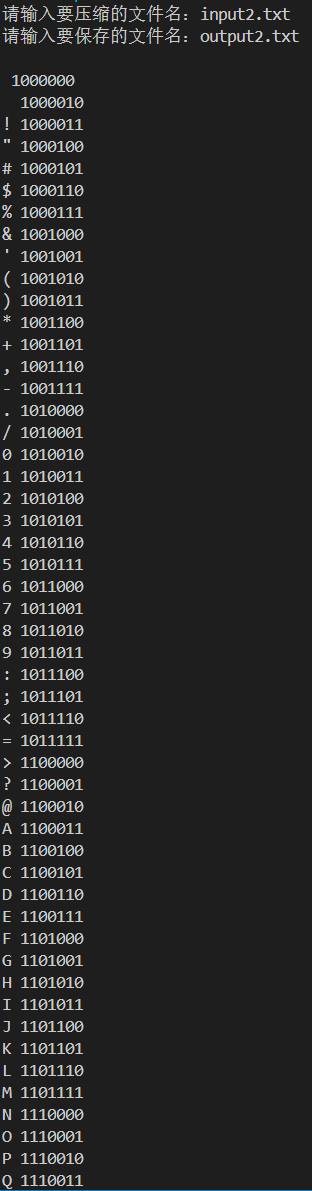
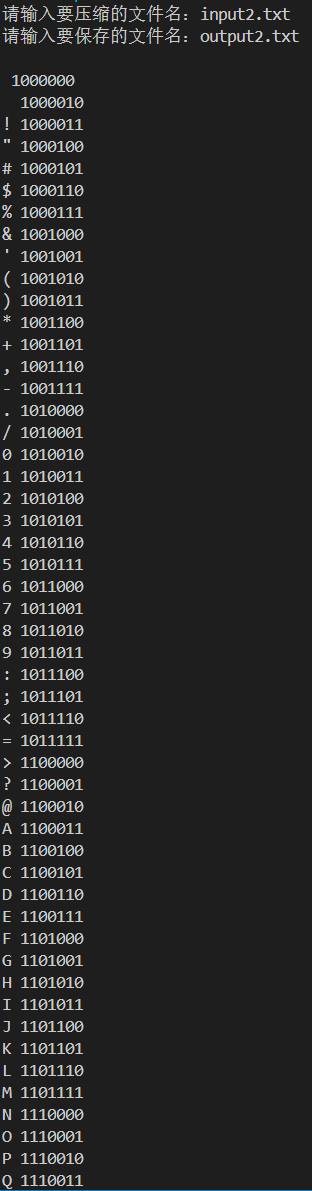
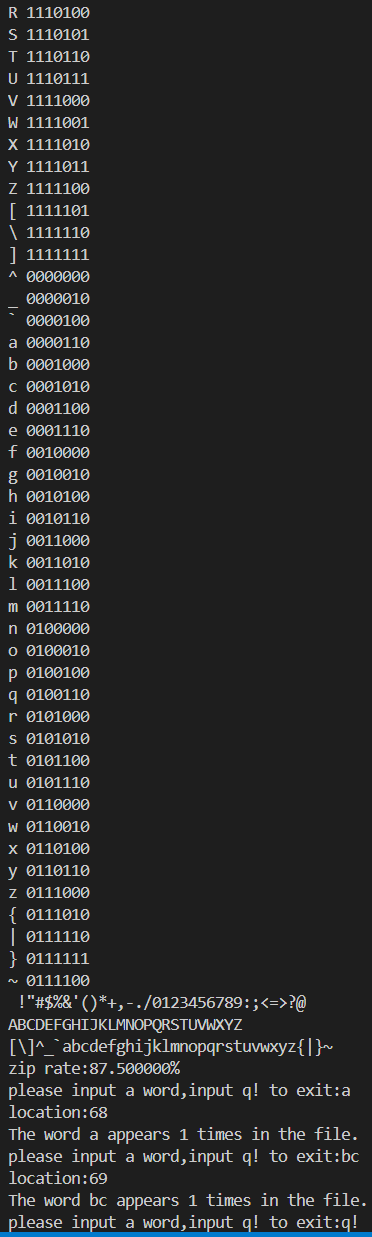


1. 运行结果截图

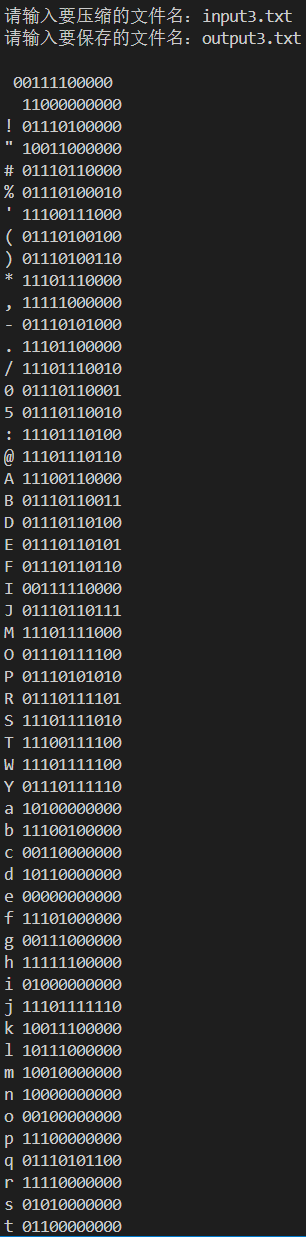
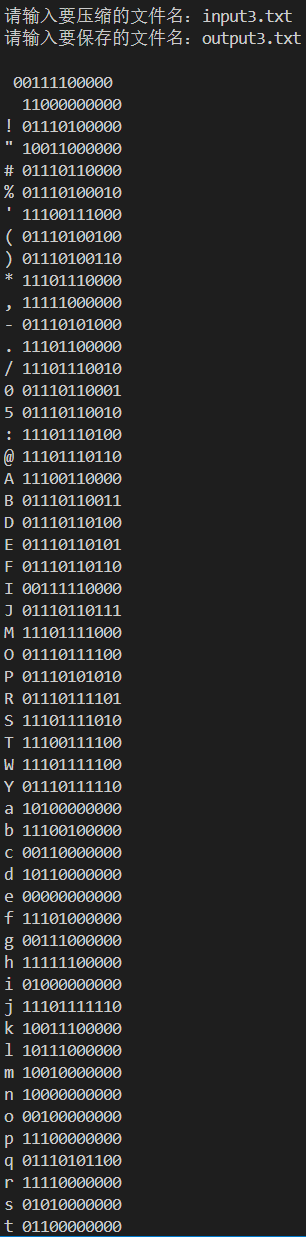
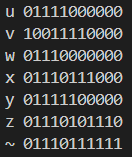


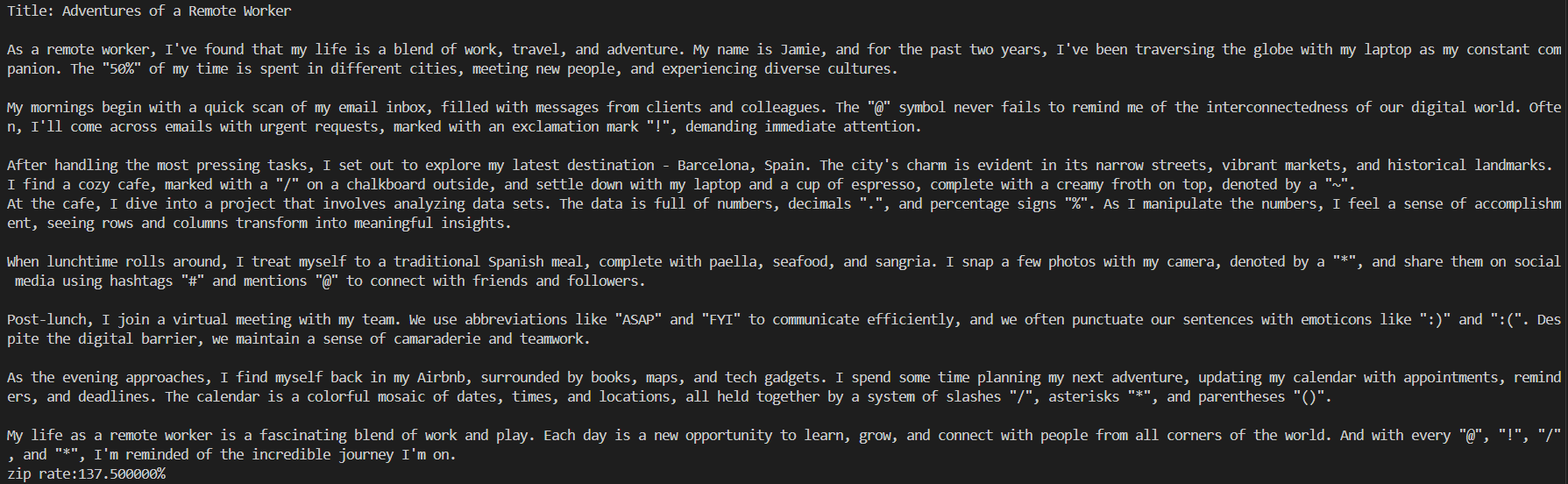
运行结果1：课本样例输出

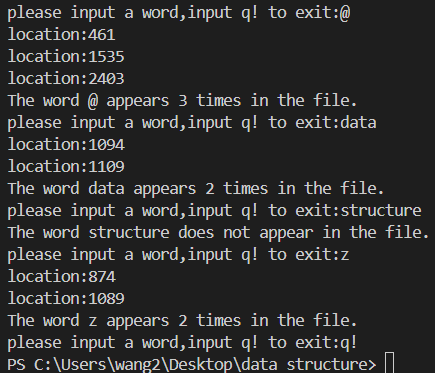




运行结果2：键盘上能输入的所有符号的情况







运行结果3：正常英文文章输入

运行结果3：较大的样例