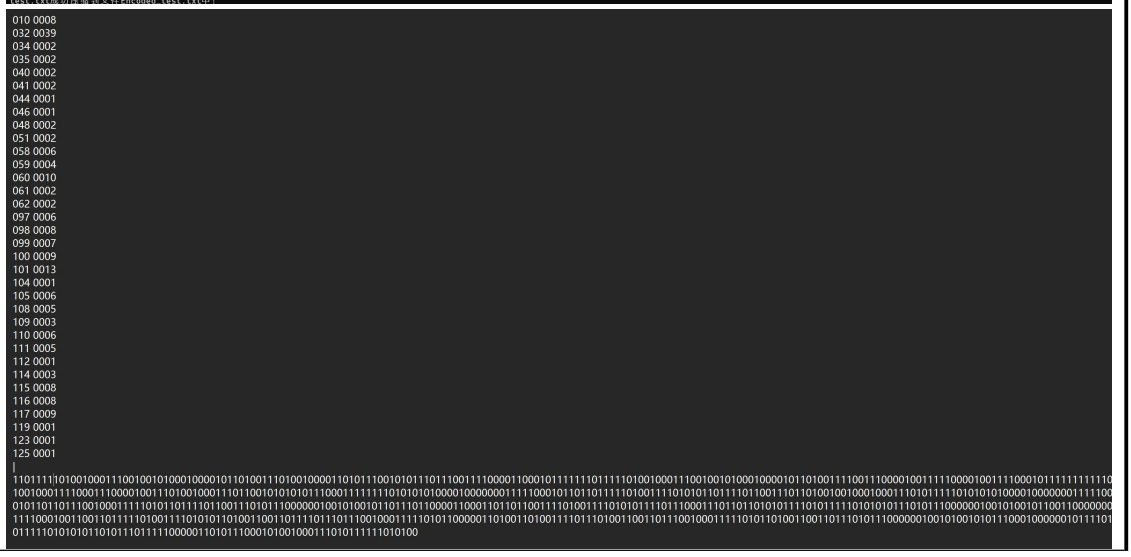
四川大学计算机学院（软件学院、智能科学与技术学院）

实验报告

学号：2023141460321 姓名：孙谦昊 专业：计算机科学与技术 班级：行政七班 第 10 周

课程名称	数据结构与算法分析课程设计	实验课时	第 3 次
实验题目	排课软件	实验时间	2024 年 11 月 12 日
实验目的和要求	实验基本要求： 1. 利用哈夫曼编码实现压缩和解压功能; 2. 编码和译码效率尽可能高。		
实验环境	12th Gen Intel(R) Core(TM) i7-1260P 2.10 GHz Visual Studio 2022 Debug x64 本地 Windows 调试器		
算法描述	哈夫曼编码（Huffman Coding）是一种贪心算法，通过构建一个霍夫曼树（Huffman Tree），为输入数据中的每个字符分配一个唯一的二进制编码，其中出现频率高的字符被分配较短的编码，而出现频率低的字符被分配较长的编码。 TreeNode 类：定义霍夫曼树的节点，包含字符、频率、父节点和子节点。 Compare 结构：定义优先队列的比较函数，确保队列顶部总是频率最小的节点。 GenerateCodes 函数：递归地为霍夫曼树中的每个字符分配编码。 DecodeCodes 函数：根据编码数据和霍夫曼树解码原始数据。 EncodeFile 函数：实现霍夫曼编码的整个流程，包括读取文件、统计频率、构建霍夫曼树、分配编码、编码数据和写入压缩文件。 DecodeFile 函数：实现霍夫曼解码的整个流程，包括读取压缩文件、解析编码数据、构建霍夫曼树、解码数据和输出解压缩后的内容 在哈夫曼编码的过程中，首先遍历输入字符串，使用一个大小为 256 的数组 arr 来统计 ASCII 字符集中每个字符的出现频率。随后构建优先队列，使用 C++的 priority_queue，配合自定义的比较函数对象 Compare，确保队列顶部总是频率最小的节点。接下来构建哈夫曼树，当优先队列中有多于一个节点时，移除两个频率最小的节点，创建一个新的内部节点作为它们的父节点，并将新节点的频率设置为两个子节点频率之和，然后将新节点重新加入优先队列。重复此过程直到队列中只剩下一个节点，这个节点就是霍夫曼树的根节点。然后为每一个叶节点分配编码，递归函数 GenerateCodes 遍历霍夫曼树，为每个叶节点（字符）分配一个编码。向左走时追加"0"，向右走时追加"1"。最后使用上一步得到的编码替换原始数据中的每个字符，生成压缩后的数据。并在接下来立即遍历原始数据，使用 map 查找每个字符对应的编码，并拼接成压缩后的字符串。 在霍夫曼解码算法的过程中。首先从文件中读取压缩后的编码数据，并根据存储的信息重新把哈夫曼树构建出来。重新构建出哈夫曼树后，从霍夫曼树的根节点开始，根据编码数据中的位向左或向右遍历树。每次到达叶节点时，输出该节点对应的字符，并返回根		

	节点。最后循环遍历编码数据，使用 DecodeCodes 函数解码每个字符，直到所有编码都被处理。并在屏幕上进行输出解码后的结果。
源程序清单	1.实验 3.cpp 主程序，调用各类函数和实现哈夫曼编码解码
运行结果	一、程序可运行功能 程序功能：1. 根据输入的待排序课程序列文件,输出一种合法的排课顺序方案。 2. 输入数据的合理性判断; 3. 用文本存储输入数据与输出各学期的课表; 4.可视化输出每学期的课表概览和每学期的可视化课表 二、程序实际运行结果



<table>
<tr><td></td><td>

```
选择输入压缩文件（E）或解压缩文件（D）：D
输入需要解压缩的文件名：Encoded_test.txt
解压缩后的内容为：
#include <iostream>
#include <cmath>

int main() {
    double base = 3.0;
    double cube = std::pow(base, 3);
    std::cout << base << " cubed is " << cube << std::endl;
    return 0;
}
```

</td></tr>
<tr><td>实验运行情况分析(包括算法、运行结果、运行环境等问题的讨论)</td><td>一、程序运行情况分析
本程序采用的算法运行出的结果与预期期望相同。正确的实现了课程的安排和课表的可视化输出。
二、运行环境讨论
本程序的编写、编译、运行和调试皆是借助 Visual Studio 2022 Debug x64 本地 Windows 调试器。若将本程序使用 DEV C++等软件上编译、运行和调试时可能报错，而这是因为各软件对于支持的 C++版本不同所导致的。例如在 Visual Studio 2022 中可以使用 nullptr，而在部分 DEV C++中仅能使用 NULL。
本程序因为采取的写作思路与事例程序有一定差别，因此代码长度显著长于事例代码。借助 Visual Studio 2022 Debug x64 本地 Windows 调试器强大的功能能够快速编译运行，而在部分 DEV C++上则速度较慢。表明程序仍有一定优化空间。
三、个人心得
此次程序的写作过程中自己付出了较多的精力，从代码的最底层设计思路和代码的写作基本上是靠自己一个人完成的（中途借鉴了下拓补排序的写法）这一次实验中综合地使用了广义表、栈、队列等形式的数据结构，使我对之前的知识点进行了复习。
当然也有很多多的遗憾。比如说真正的压缩文件一般不能通过直接点击的方式打开查看到其中的内容，我压缩到的压缩文件也是一个一个的字符，其实压缩率并没有达到非常理想的效果。
总体而言，完成这样一个程序是非常让我自豪的！
在编写程序过程中的相关版本代码发布已经在 Github 中，链接为：https://github.com/StrayerSQH/Learning/tree/main/%E6%95%B0%E6%8D%AE%E7%BB%93%E6%9E%84%E4%B8%8E%E7%AE%97%E6%B3%95%E5%88%86%E6%9E%90%E5%AE%9E%E9%AA%8C%E8%AF%BE/%E5%AE%9E%E9%AA%8C4</td></tr>
<tr><td>指导老师评议</td><td>成绩评定：　　指导教师签名：</td></tr>
</table>