**系统测试**

1 测试目标

测试压缩程序的压缩、解压功能在处理不同类型的文件时是否可以正常工作，以及压缩的效率。

2 测试方法

黑盒测试法：也可以被叫做功能测试以及数据驱动测试，它与白盒测试完全相反，与系统的逻辑结构，或者路径完全无关，它只是以程序为主测试程序的功能以及性质，测试程序的执行，是否能够将系统的功能进行实现，对程序的准确率进行检测。

3 测试用例

1、压缩功能测试

测试目的：压缩功能的有效性。测试前提是文件大小大于1kb，此测试共设计了 2 个用例，如下表1，表2所示。

表1测试用例 1

|  |
| --- |
| 编号： compress01 |
| 项目： 压缩功能用例测试 |
| 内容： 压缩10kb大小的markdown文件 |
| 前提条件：文件大小大于1kb |
| 测试步骤： |
| 1.运行程序选择压缩对并输入文件路径地址 |
| 2.对文件进行压缩 |
| 预期结果：压缩成功，压缩后文件大小为8kb。 |
| 结果和结论：一致。 |

表2测试用例 2

|  |
| --- |
| 编号： compress02 |
| 项目： 压缩功能用例测试 |
| 内容：压缩大小为9kb的文本文件 |
| 前提条件：文件大小大于1kb。 |
| 测试步骤： |
| 1. 运行程序选择压缩并输入文件路径 |
| 2. 对文件进行压缩 |
| 预期结果：压缩成功，压缩后文件大小为5kb。 |
| 结果和结论：条件受限，借阅失败。 |

2.解压功能测试

测试目的：测试程序是否能正确解压被压缩过的文件。测试前提是被解压的文件是被本程序压缩的，此测试共设计了2 个测试用例，如下见表3，表4。

表3测试用例 3

|  |
| --- |
| 编号： decompress01 |
| 项目： 解压功能用例测试 |
| 内容： 对压缩过的md文件进行解压 |
| 条件：被解压的文件是被本程序压缩过的，即后缀为huf |
| 步骤： |
| 1. 运行程序选择解压后输入文件路径 |
| 2. 对文件进行解压 |
| 预期结果：解压成功 |
| 测试结果和结论：一致 |

表4测试用例4

|  |
| --- |
| 编号： decompressa02 |
| 项目： 解压功能用例测试 |
| 内容：解压被压缩过的文本文件 |
| 前提条件：系统存在已知读者及相关信息。 |
| 测试步骤： |
| 1. 管理员对读者借阅卡、以及图书编号进行扫描 操作 ； |
| 2. 扫描成功，信息读取成功； |
| 3.点击归还。 |
| 预期结果：归还图书成功。 |
| 测试结果和结论：一致。 |

3.对Huffman压缩性能的分析

Huffman编码效果好的文件类型

1. 符号分布不均的文件：Huffman编码在一些符号出现频率明显高于其他符号的文件中效果特别好。例如，在英语文本文件中，某些字母如 'e', 't', 'a', 'o' 的出现频率比 'z', 'q', 'x', 'j' 高得多。霍夫曼编码利用这种不均匀的频率分布来实现较好的压缩效果。

2. 有限色调的灰度图像：某些像素值比其他值更常见的灰度图像中，Huffman编码也能表现得非常好。

霍夫曼编码效果差的文件类型

1. 符号分布均匀的文件对于每个符号出现频率大致相等的数据，Huffman编码效果较差。

2. 已经压缩过的数据：已经使用其他算法（如 JPEG 图像、MP3 音频文件或 ZIP 文件）压缩过的文件通常没有多少冗余可以让Huffman编码利用。尝试对这些文件进行Huffman编码通常会导致效果不佳。

3. 非常小的文件：对于非常小的文件，Huffman树本身的开销（需要与压缩数据一起存储，以便解压缩）可能超过任何压缩增益。