PROJECT 1

File Compression Using Huffman Coding

Introduction

File compression is an important part in saving disk storage and helping reducing transmission time. Generally, there are two types of compression mechanisms - lossy compression and lossless compression. As their name implies, the lossy compression, which is often used for media data, would lead to content loss after recovering the compressed file, while lossless compression can recover exactly the original file. The compression format often seen in computers such as .zip, .rar belongs to lossless compression.

In this project, a tool with compression and decompression is required to be implemented by using Huffman Coding. After your submission, a face-to-face interview is also required, so please remember to make an appointment for your interview after finishing.

Requirements

- Program Language: Java
- An executable program with
  - compression function using Huffman Coding
  - decompression function
  - user-friendly GUI
  - support both folder and single file
- A user manual
- A development document which should includes
  - your design and implementation in detail
  - performance test results such as the compression rate, compression time and decompression time for given test cases.
  - problems encountered in this project and your solutions
- *Optional*
  - compare the compression rate and running time of your implementation with other standard compression tools (WinRAR, 7Zip, etc)
  - write down the techniques you used in optimizing and show the comparison results

Grading

Points	Percentage
Single file compression & decompression	30%
Folder compression & decompression	30%
UI design & User Manual	15%
Performance (Time & Rate)	15%
Development Documentation	10%
Optimization & Comparison	10% (bonus)

Submission

Please put all your source codes, required documents, and an executable file into a zip file with file name *StudentID-Name.zip* and upload it to FTP server **before 2016/10/20 23:59 (GMT+08:00)**.

Besides, a face-to-face interview is also required to grading your project, so please remember to make an appointment for your interview after finishing. The deadline for interview is **2016/10/27**.

No Plagiarism! No Extension!

Tips

- General steps for compression: (just in case, it's suggested to think by yourself ^^)
    - read in the file and count the character frequency
    - build a huffman tree to encoding characters
    - store the file using huffman coding (along with the tree)
- Think carefully about decompression
    - how to store the huffman tree so that it can be recovered efficiently in decompress
    - how to recover characters since the length of their encoding is not fixed

- Many file types are already compressed, like JPEG and MPEG. You can have a try doing compression with standard tools for those files to see what will happen.
- The running time is somewhat relevant to the computer hardware, so please remember to include your computer's configuration along with your efficiency test.
- Avoid concatenation or subtraction of strings can help reduce the running time.
- Take care of EOF in compression and decompression.
- Take care for the cases of empty file and empty folder.
- If encountering any problem, try to think on your own at first and then search for solutions online.
- Any ambiguity in this document, please contact TA.

Sisi Li (16212010014@fudan.edu.cn)

28 September 2016