Project 1 - Zip Tool with Huffman Coding

WenqiangRuan & Chuanwang Wang

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 as soon as possible after finishing.

Submission

Please put all your **source code, required documents, and an executable file** into a zip file with file name **StudentID-Name-PJ1.zip** and upload it to https://wss.pet/s/3lngy7f8eco (password: 3854) before **2020/11/1 23:59 (GMT+08:00)**.

The structure of your zip file should be:

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 2020/11/4**.

Grading

Required: in Java (JDK 8, JDK 11), error message encapsulation.

> 100 points in total.
>
> 55 points for compression and decompression
>
> 15 points for performance (speed and ratio)
>
> 15 points for UI
>
> 15 points for documentation (Manual, DevDocs, Comparison)
>
> 0 for **essential** executable program. jar or any other format.

Content	Points
Empty single file (0 K) compression & decompression	5
Small~Large single file (1 K ~ 10 M) compression & decompression	5
X (extreme) Large single file (1 G) compression & decompression	15
Empty folder compression & decompression	5
A folder, some small~large single files (may be empty, *not x-large*) compression & decompression	5
A folder, some sub-folders (may be empty, may have sub-folders), **some small~large single files** (may be empty, *not x-large*) compression & decompression	5
A folder, some sub-folders (may be empty, or have sub-folders, *x-large* files), **some single files** (may be empty, *x-large*) compression & decompression	15

- **Compression speed (7.5 points)**

  Explicitly show the average compression speed,

  Zip method implemented by Java runs at about 18M/s (i7-8700, 24G, 1T HDD)

Speed	Points
[0, 0.5M/s)	0
[0.5M/s, 2.5M/s)	2.5
[2.5M/s, 7.5M/s)	5
[7.5M/s, $+\infty$)	7.5

- **Compression ratio (7.5 points)**

  Explicitly show the compression ratio.

  - Related to your ranking of compression ratio
- **UI design (15 points)**
  - Easy to use
  - Messages (average speed, compression ratio, time used, start time, end time etc.)
- **User Manual (5 points)**

  PDF file.

  - All functions, and how to use these functions (**3 points**)
  - All possible problems, and how to handle these problems (**2 points**)
- **Development Documentation (5 points)**

  PDF file. One or two pages.

  - The insight
  - Problems encountered
- **Comparison (5 points)**

  CSV file.

  - WinRAR (**2.5 points**)
  - HaoZip **(2.5 points**)

Tips

- Think carefully about
    - 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.
- Be care of EOF in compression and decompression.
- Be care of empty file and empty folder.
- If encountering any problem, try to think on your own at first and then search for solutions online.
- If your find any ambiguity in this document, please contact TA.