WRITEUP ASGN6

Cruzid: acristea

March 2023 - CSE13S

1 Harnessing Skills New Concepts

This assignment asked us to learn a new compression and decompression method: Lempel-Ziv Compression. Having this being my first time working with compression and decompression, I had to dedicate a lot of time to digest the concept and understand the full process.

One method that I have learned to fully embrace in all aspects of any assignment was watching an example walk-through on YouTube (which has been a big part of how I go about understanding new concepts or concepts that I typically don't digest well after lectures).

For the implementation of the encoding and decoding functions, we had to work with structs such as Word, Trie, WordTable, TrieNode, etc. It was a lot of work just trying to be able to work with so many variable types and the contents of the struct themselves (especially when they were pointers to other structs). After this assignment, I would say that my understanding of structs and how to manage memory such as allocating and freeing has definitely been refined and strengthened.

Because this assignment took a similar route to assignment 5 (constructing and deconstructing), my overall comfort in making those executables increased. I would say that the main area I struggled in was the IO file because I hadn't had too much experience working with bits (which I go on about later as well).

I think that having to juggle so many files, code, and executables enabled me to grow as a coder. Just having to deal with a lot of segfaults, utilizing valgrind, and identifying which file it was in became almost natural. The point is, the biggest thing I learned was perseverance.

2 Functionality of LZ78 Compression and Decompression

LZ78 uses a trie to keep track of its symbols and their corresponding code. As our program goes through its symbols, it adds them to the trie. If a symbol has been used already, it checks the symbol before it and adds it almost like it's their child. It takes the code of its predecessor but keeps the symbol. This meant I had to learn how a trie works, and how it's implemented in compression. The most important factor I had to learn was bit masking. We had to iterate through bits, and bytes, and check to see what their bit value is. Working with buffers was hard at first, as it was hard to keep track of where you are in the buffer, or even if you were filling the buffer up correctly. Eventually, after writing out the buffer by hand and ensuring all my bits were written correctly, I started to complete the assignment. LZ78 decompression uses a word table and goes through its symbols one branch at a time until it reaches the end, which then returns our original texts. Write Pair and Read Pair were the hardest parts, as although it passes the tests, it doesn't mean it encodes/decodes it correctly. That was something I struggled with a lot, as I was iterating through the buffer fine, but not writing/reading to them correctly. Eventually, I learned the correct way.

3 How the Efficiency of Compression Changes With Entropy

According to Wikipedia, entropy compression is an information-theoretic method for proving that a random process terminates. In a set of data, it is related to the amount of information that it contains which is directly related to the amount of compression achieved. Essentially, the entropy of a file equates to a measure of the "randomness" of the data in a file.

That being said, the relationship between entropy and the efficiency of compression shares a positive correlation. With that, we know that as compression increases in efficiency, entropy increases as well. In other words, the data is less predictable than it was prior when it is more efficient.

If we were to look at it the other way around, if the efficiency of compression were to decrease, entropy would decrease as well. For example, when my compressed file size was 122 bytes, and my uncompressed file size was 130 bytes, my compression ratio was 6.15 percent. If I increased the number of bytes I had in my uncompressed file, to 246 bytes, my compressed file becomes 208 bytes, and the ratio has risen to 15.45 percent.

4 Conclusion

In conclusion, this assignment taught me a lot about the functionality of compression and decompression of data files, specifically speaking the LZ78 method. In terms of my C coding knowledge, I have been able to hone and master certain skills such as debugging and creating my own separate test files. Although the assignment seemed daunting at first, the more hours I put into it, the more I understood the process and how we needed to code it. I know a lot of fellow students had trouble with test cases, but that's because they only test specific cases, and don't call every function, which I had to explain to quite a few of them. This was the first assignment I helped others during section and it made me realize how far I had come learning and coding in C.