Simple Encoding and Decoding in C

Logan Day

January 19, 2024

1 Reflection

This project was a fun way to review file io in c. Judging by the sample output for the test file given on canvas, it looks like my program properly encodes the input data. Also, when I plug the encoded data back into my decoding program, the original data is produced. I wrote a bash script in order to test the encoding and decoding efficiency of the program. The script appends 10,000 bytes of lorem ipsum text to an input file each time it runs. I also modified my program to print the time it took to run the encoding/ decoding method, and the file length in bytes to an output file, then plugged this file into google sheets to generate charts. Both the data for encoding and decoding appears to be linear, so I suspect my methods are O(n) with respect to the input size.

```
#!/bin/bash

gcc filesec.c

rm output.txt
rm rec_test.txt
touch rec_test.txt
touch output.txt

for ((i=0; i<=100; i++))
do
    cat "lorem.txt" >> rec_test.txt
    ./filesec -d "rec_test.txt"
done
```

Figure 1: Bash script used to run filesec on increasingly large input sizes. 10,000 bytes of randomly generated lorem text is added to the input file each time.

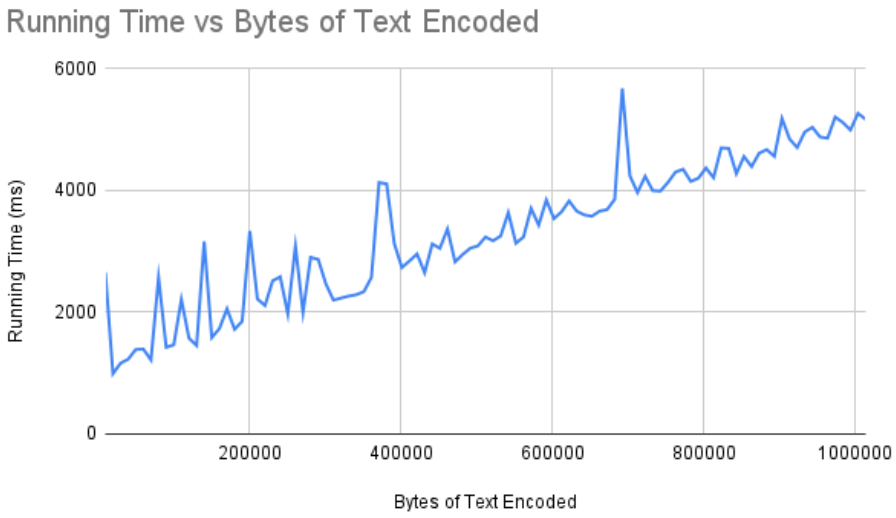

Figure 2: Running time for encoding (ms) charted against length of input over 100 trials. Each trial, the input file length was increased by 10,1000 bytes

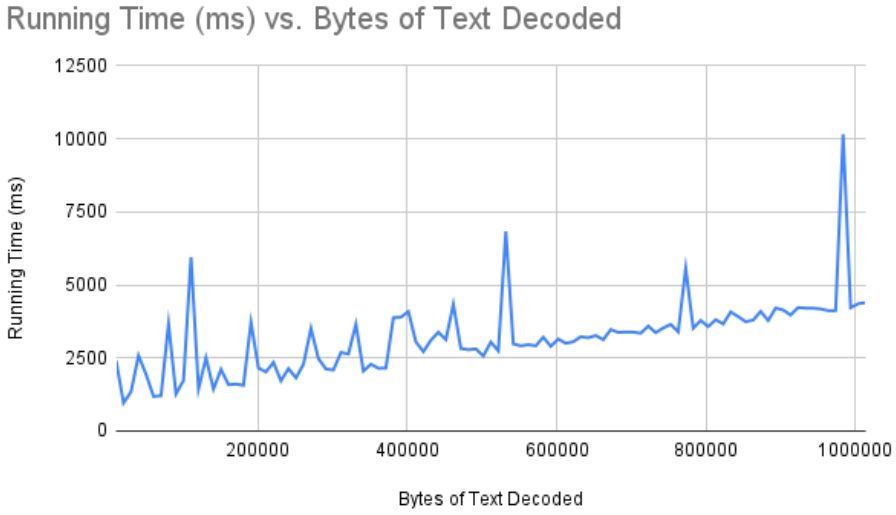

Figure 3: Running time for decoding (ms) charted against length of input over 100 trials. Each trial, the input file length was increased by 10,1000 bytes