

Caleb Obeng

I pledge my honor that I have abided by the Stevens Honor System

### **Text2bin:**

Approach: I looped through each line of the u data file. For each line split the lines using strtok. Once I had the pieces of each line, I converted each of the parts into their corresponding byte numbers. Lastly I wrote all of these byte numbers as binary to the output file using fwrite.

Real time 100k: 0.47

Real time 1 Million: 4.17

### **Bin2text:**

Approach: I used fseek and ftell to get the total number of bytes I would need to loop through. I stored that value in a variable to let me know when I've reached the end of the file. I looped every 13 bytes, using fread and fprintf to read the bytes in sets of 2, 2, 1, and 8. I used bit shifting combine the bytes and write the whole number.

Real time 100k: 1.02

Real time 1 Million: 6.22

### **Bin2Indexed:**

Approach: First, I loop through all of u item and get all of the offsets for each line, then stored it in a malloc array, realloc if needed. Then I found the total number of bytes that I would need to loop through in the input file. Then I just wrote all of the binary numbers to the output, replacing the second column with then offset values from the array I created.

Real time 100k: 1.50

Real time 1 Million: 5.30