## **Prompts for Question I and II:**

- Upload Assignment2 file to ChatGPT and ask it "Can you solve this assignment for me?"
  - Then, ChatGPT asks: "Would you like help coding these solutions or any specific part of this assignment?"
- 2. Yes, please code the solution.
- Text Processing

First function for counting lines, words, characters, and uppercase characters

```
#include <stdio.h>
                                                             // Count words
#include <stdlib.h>
                                                             if (isspace(ch)) {
                                                                 if (in_word) {
void analyze_file(const char* filename) {
                                                                     in_word = 0;
   FILE *file = fopen(filename, "r");
                                                                     words++;
   if (file == NULL) {
                                                                 in_word = 1;
   int lines = 0, words = 0, chars = 0, upper_case = 0;
   int in word = 0;
                                                         // Account for EOF line and the last word
   while ((ch = fgetc(file)) != EOF) {
                                                         lines++;
       if (ch == '\n') {
                                                         if (in_word) {
          lines++:
                                                             words++;
       // Count alphanumeric characters
                                                         printf("Lines: %d\n", lines);
       if (isalnum(ch)) {
                                                         printf("Alphanumeric characters: %d\n", chars);
          chars++:
                                                        printf("Uppercase letters: %d\n", upper_case);
                                                         printf("Words: %d\n", words);
       if (isupper(ch)) {
                                                         fclose(file);
          upper_case++;
```

Second function for converting characters to uppercase characters

```
void convert_to_upper(const char* input_filename, const char* output_filename) {
   FILE *input_file = fopen(input_filename, "r");
   FILE *output_file = fopen(output_filename, "w");

   if (input_file == NULL || output_file == NULL) {
        printf("Error opening files.\n");
        return;
   }

   char ch;
   while ((ch = fgetc(input_file)) != EOF) {
        fputc(toupper(ch), output_file);
   }

   fclose(input_file);
   fclose(output_file);
}

int main(int argc, char *argv[]) {
   if (argc != 2) {
        printf("Usage: %s <input_file>\n", argv[0]);
        return 1;
   }

   analyze_file(argv[1]);
   convert_to_upper(argv[1], "UPPER");
   return 0;
}
```

## I. Pointers

```
#include <stdio.h>
#include <stdlib.h>
int main(int argc, char *argv[]) {
    if (argc != 2) {
        printf("Usage: %s <memory_size>\n", argv[0]);
        return 1;
    // Convert argv[1] to integer using atoi
    int size = atoi(argv[1]);
    if (size <= 0) {
        printf("Invalid memory size.\n");
        return 1;
    // Allocate memory using malloc
    char *memory = (char*) malloc(size);
    if (memory == NULL) {
        printf("Memory allocation failed.\n");
    char pattern[] = "ABCDEFGHIJKLMNOPQRSTUVWXYZ123456789";
    int pattern_size = 35;
    for (int i = 0; i < size; i++) {
        memory[i] = pattern[i % pattern_size];
    // Print the generated string
    printf("Generated String: %.*s\n", size, memory);
```

```
// Process memory with an int pointer and while loop
int *int_ptr = (int*) memory;
int bytes_processed = 0;

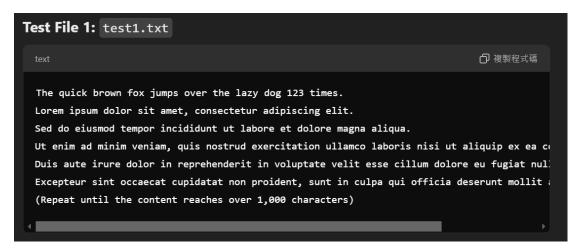
while (bytes_processed + sizeof(int) <= size) {
    printf("%d : %d\n", *int_ptr, *int_ptr - 1);
    int_ptr++;
    bytes_processed += sizeof(int);
}

// Free allocated memory
free(memory);

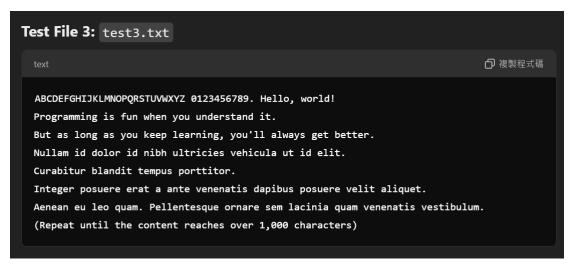
return 0;
}</pre>
```

## Prompts for generating 3 test text files:

"For Text Processing, you must create at least 3 test text files, each at least 1,000 characters, to verify the program's correctness. Can you generate these three files with a line no more than 80 characters?"







## **Prompts for generating TESTPLAN:**

Details is in the TESTPLAN.pdf.