EECE.2160: ECE Application Programming
Key Questions
File, Character, and Line I/O (Lec. 31-34)

QUESTIONS:

1. Explain the use of the `fopen()` function.

2. Explain the use of the `fclose()` function.

3. Explain how `fscanf()` and `fprintf()` are used for formatted file I/O.

4. Explain how `fread()` and `fwrite()` are used for unformatted I/O.

5. Describe the standard input and output streams.

6. Describe how to test that a program has reached the end of a file or encountered an error.

7. Explain the functions used for character input.

8. Explain the `fgets()` function for line input.

EXAMPLES:

Complete the below program to:
- Read three integers from file `myinput.txt`
- Determine the sum and average of those values
- Write the original values, sum, and average to file `myoutput.txt`.

```
int main() {
  int v1, v2, v3, sum;   // Input values and their sum
  double avg;            // Average of input values
  FILE *fpIn, *fpOut;    // File pointers

  // Open input file and exit if error




  // Open output file and exit if error




  // Read input data, find sum & average, and print results




  // Close files


}
```

Show the output of each of the following short programs:
a. Input: **Test Input   1   23 4 5**

```c
void main() {
    char c;
    char buffer[50];
    int i, n;
    i = 0;
    while ((c = fgetc(stdin)) != '\n') {
        if (c != ' ') {
            buffer[i++] = c;
        }
    }
    buffer[i] = '\0';
    printf(buffer);
}
```

b. Input:
Test1
Test 2
abcdefghijklmnopqrstuvwxyz
This is a test of the fgets() function

```
void main() {
   char str[25];
   int i;
   for (i = 0; i < 5; i++) {
      fgets(str, 24, stdin);
      strcat(str, "\n");
      printf(str);
   }
}
```

c. Input:
1024Some other stuff
```
void main() {
   char c;
   char buffer[50];
   int n = 0;

   // isdigit in <ctype.h>
   while (isdigit(c = getchar())) {
     n = n * 10 + (c - 48);    // Hint: '0' = 48   }
     // (ASCII value)
   ungetc(c, stdin);
   fgets(buffer, 50, stdin);

   printf("n = %d, n * 2 = %d\n", n, n * 2);
   printf("buffer = %s\n", buffer);
}
```