

EECE.2160: ECE Application Programming

Summer 2017

Lecture 13: Key Questions

June 19, 2017

1. Describe how to maintain a sorted linked list.

2. Write each of the following functions:
 - a. Adding an item to a sorted linked list
 - Use **addNode()** as a starting point
 - Instead of adding node at beginning, find appropriate place in list and then add
 - Function should return pointer to start of list after it has been modified

```
LLnode *addSortedNode(LLnode *list, int v) {
```

```
}
```

- b. Finding an item in a sorted linked list
- Use **findNode()** as starting point—should perform same operation, but more efficiently
 - Function should return pointer to node if found
 - Return NULL otherwise

```
LLnode *findSortedNode(LLnode *list, int v) {
```

```
}
```

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. **Example:** Write a 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`.

- 6

7. Describe the functions used for character I/O.

- 8

10. **Example:** Show the output of each of the following short program.

a. Input: **Test Input** **1** **23 4 5**

```
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';
    fputs(buffer, stdout);
}
```

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");  
        fputs(str, stdout);  
    }  
}
```

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);
}
```