Programming Studio #11

ECE 190

Programming Studio #11

- Topics this week:
  - File I/O using <stdio.h>
- In Studio Assignment
  - ToUpper in C with file I/O

Announcements

- MP4.2 due next wednesday!
- Exam 2 grades are posted
  - Make sure you attend programming sections if required!

Opening a File

- #Include <stdio.h>
- File *file;
  - Declare file pointer
- file = fopen("filename", char *mode);
  - if fopen fails, it returns NULL
  - "r" = readonly, "w" = writeonly
  - "a" = append, file is created if necessary
  - "r+" = read/write, file must exist
  - "w+" = read/write, file is created if necessary

Where does it Read/Write?

- Files are read/written to current working directory (unless a path is specified)
- When you open a file, it starts from beginning of file (offset zero)
- When reading or writing, the offset is incremented based on the size of the read or write
- When writing to file, data at offset is overwritten if it previously existed. Once you overwrite all previous data, it appends.

Modes

	Can Read	Can Write	File Must Exist	Clears File/ Creates Empty File
"r"	Y	N	Y	N
"w"	N	Y	N	Y
"a"	Y	Y	N	N
"r+"	Y	Y	Y	N
"w+"	Y	Y	N	Y

- "a", append always writes data to end of file (never overwrites)
- File must exist means if it doesn't, fopen return NULL
- Clears File/Create empty file
  - If file exists, **ALL DATA IS REMOVED**
  - An empty file **WILL** be created

Basic File Read/Write

- Works like standard I/O
  - scanf -> fscanf
  - printf -> fprintf

```
/* writing to file */
FILE * file = fopen("file.txt", "w");
int first_int = 1;
fprintf(file, "%d", first_int);

... // close file here

/* reading from file */
FILE * file = fopen("file.txt", "r");
int first_int;
fscanf(file, "%d", &first_int);
printf("The first number is %d", first_int);
```

"file.txt" contents before write:
8
10
67
100

"file.txt" contents after write:
1

Console prints:
The first number is 1

Close File and End of File

- int feof(FILE *file)
  - returns non-zero when the end of the file has been reached

```
int nums[100];
FILE* file = fopen("list.txt", "r");
if(file != NULL)
{
        int i;
        for(i = 0; i < 100 && !feof(file); i++)
                fscanf(file, "%d", &nums[i]);
        for(; i < 100; i++)
                nums[i] = 0;
        fclose(file);
}
```

Mini-program

- Implement your ToUpper program from MP1 in C
  - Remember, should take every lowercase letter and capitalize it
- Read in input filename from user using scanf
- Overwrite input file with modified version
  - While still testing/debugging, you will probably want to write to a temporary file, in order to avoid needing to recreate the input
- You may assume the input file has no more than 100 characters
- Hint: you should read in and buffer all the characters in one phase, and write the modified characters back into the file in another phase