# CS211: Programming Assignment 2:

# Functions operating on 2D Arrays and Strings

DUE: Wed, Feb 16 by 11:59PM

You are part of a startup called ASCII-unlimited. You plan to dominate the market in character based graphics for gaming! Your team has identified a set of 13 fundamental operations that are likely to be useful for many of the games you ultimately produce. Accordingly, you are developing a library of C functions which implement these operations.

For this assignment you will write these 13 functions. There are 11 functions operating on square dynamically allocated 2D arrays of characters and 2 operating on C strings.

The table below lists the function prototypes and assigns each an estimated difficulty level on a scale of 1-3 (just for rough guidance -- you may or may not end up agreeing with the scores…)

| **FUNCTION** | **ESTIMATED DIFFICULTY (1-3).**  **1: less challenging**  **3: more challenging** |
| --- | --- |
| char \*\* alloc\_square\_mtx(int n); | 1 |
| void free\_square\_mtx(char \*\*m, int n); | 1 |
| void pop\_mtx\_alpha(char \*\*m, int n); | 1 |
| void display\_mtx(char \*\*m, int n); | 1 |
| void swap\_rows(char \*\*m, int r1, int r2, int n); | 2 |
| void swap\_cols(char \*\*m, int c1, int c2, int n); | 2 |
| void rotate\_right(char \*\*m, int n); | 3 |
| void floating\_boulders(char \*\*m, int n); | 1.5 |
| void mountains(char \*\*m, int n); | 2 |
| void sink(char \*\*m, int n); | 2.5 |
| void gravity(char \*\*m, int n); | 3 |
| void str\_trim(char \*s); | 3 |
| int str\_search(char \*s, char \*pattern); | 3 |

# What To Do

1. Download the files prog2.h and prog2.c from the src directory (from the same location you found this handout).
   * **prog2.h** is a header file which you are not supposed to modify.[[1]](#footnote-0) It contains the function prototypes and, more importantly, the detailed description of how the functions are supposed to behave.
   * **prog2.c** is where you will do your work -- completing the implementation of the functions. It is pre-populated with function “stubs” along with a **main** function where you can write your test code to exercise your implementations.

However, do not remove

1. Read through the “banner comments” above each function prototype in prog2.h. Maybe print out a hardcopy for reference. These comments explain the expected behavior of the functions -- i.e., they are the “specs” for this assignment.
2. Start working on the functions!
   * Test each function (and debug it accordingly) before moving on to the next.
   * Start with the easier functions.

# Submission

You will just submit your completed implementation file prog2.c through blackboard. You do not need to submit prog2.h because it should be unchanged.

# Some Things to Remember

* You are NOT modifying prog2.h. All of the code you write will be in prog2.c.
* C strings are "sentinel-terminated" (the “end of string” character ‘\0’ marks the end of the string).
* The functions operating on 2D arrays work on 2D arrays of characters. These are *not* strings; nor are they arrays of strings.

1. To clarify: you may not alter any of the contents of prog2.h; however, if you want to add additional functions for debugging purposes (which can be invoked from a "driver" program), that is acceptable. Key concept: any modifications must be strictly "additive" [↑](#footnote-ref-0)