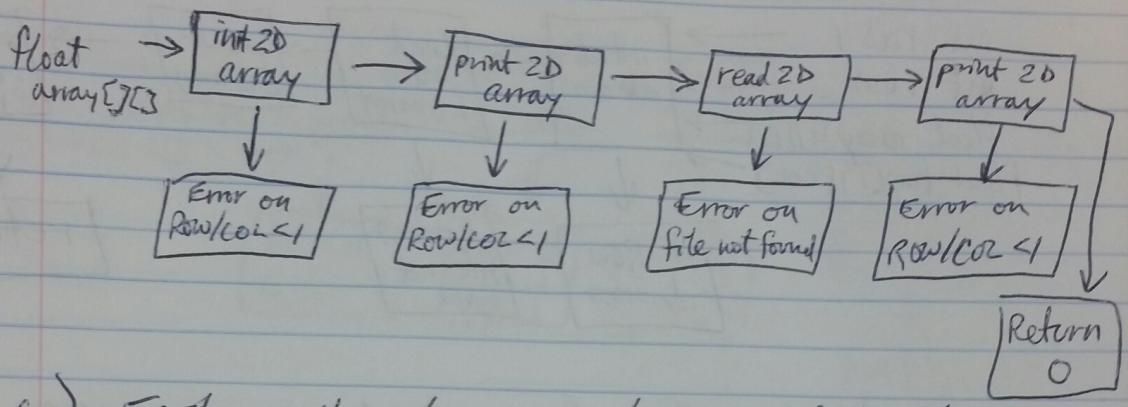
Shane Ryan
EPSM

1.) a.) Purpose - initalize 2D array + print. Then read file into it + print.

b.)



c.) First, a two dimensional array of float must be created and passed to the function init2D array. There its values will be set to zeros by a for loop going through every entry. Then the array is passed to print2D, which will step through a for loop to print out array values (0's). Read2D array can now be called, taking a file name, opening a stream, and using fscanf() to read a csv into the array. Finally, the array goes to print2D array again to be printed.

d.) float [ROW][COL]

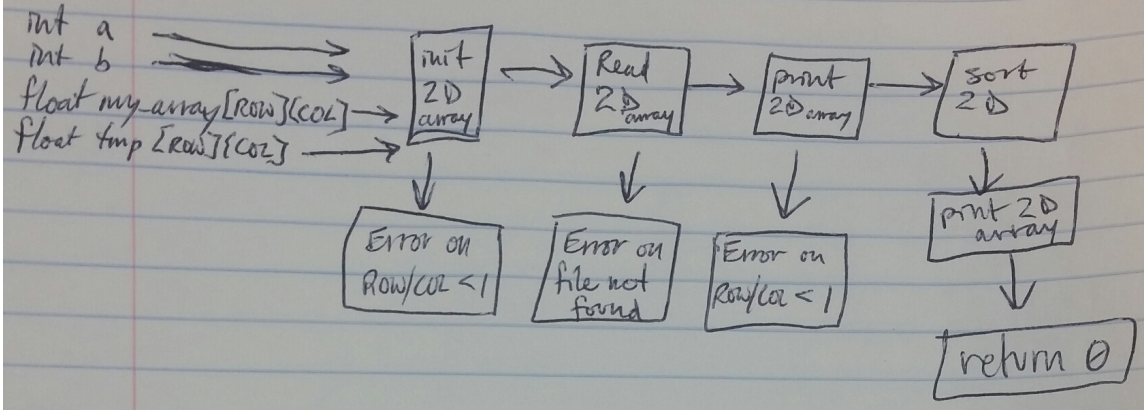
The loop below should be very useful for this program

Example stepping loop:

```
for (i=0; i<ROW_SIZE; i++) {
    for (j=0; j< COL_SIZE; j++) {
        fscanf or printf or set to zero;
    }
}
```

3.) a.) Purpose: Sort a 2D array by dollar earned, from highest to lowest, corresponding to data read in from a .csv.

b.)

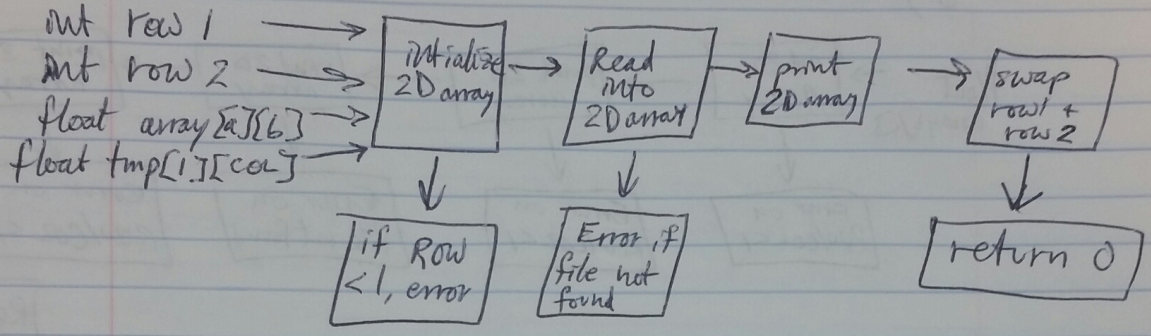


c.) The new function Sort 2D Float array will compare the value of money for one row with the next row, and if the latter is smaller, the two row values will be passed to Swap Two Rows to be switched. This will keep looping until no rows are swapped, indicating the list is properly sorted.

d.)

```
sort2DFloatArray(my_array)
while state==1 { { for(i=0;i<ROWSIZE;i++) {
    if (my_array[i][2] > my_array[i+1][2])
        swap(i, i+1); counter ++;
    else
        if counter > 0
            state = 1;
            counter = 0;
        else
            state = 0; }
return 0;
```

2.) a.) Purpose: Do the same as Prog1 (initialize 2D array, read file into array, print array) and then swap two rows of the array and reprint.



c.) Everything from prog1.c carries over into prog 2, so first → (initalize, read into array, print) (prog1), then pass the int row1 and int row2 options to swapTwoRows along with the filled array, and then print again. The new function will store the first row's data in tmp variables, set the 1st row equal to second row data, and 2nd row data equal to 1st row temp data.

d.)
```
#include <lab8.h>

{
    call initalize
    call Read
    call print
    call swap
    call print
}
```

(Switch 2 terms) → (loop also)

```
swapTwoRows(a, b, my_array)
{ for((i=0); i<COL-SIZE; i++) {
    my_array[a][i] = tmp[0][i]
    my_array[a][i] = my_array[b][i]
    my_array[b][i] = tmp[0][i]
}
return 0;
}
```