

Excess Functions

Passing a 2D-array

To pass an array to a function, you have to put the brackets in the prototype and definition. However, you don't need them when you call the function.

Prototype:

```
int arrayStuff(int myIntArr[]);
```

When you want to pass a 2D-array to a function, you need the brackets and the width (second dimension). Just like with 1D-arrays, you don't need any brackets when you call the function.

Prototype:

```
int moreArrayStuff(int my2DArray[][SIZE]);
```

Program

For this program, I want you to augment the functionality from Lab6. Note that I will be asking you to use many functions. It will probably be more than are necessary, but the purpose of this lab is to use functions. This time your program will prompt the user what to print, fill the 2D-array appropriately, print what they want, and then repeat until they enter a 'q'. Follow these specs:

1. Prompt the user for the empty and fill characters (like in Lab6).
2. Prompt the user for what they want to do:
 1. If they enter 'x', fill the 2D-array with an X (like in Lab6).
 2. If they enter '+', fill the 2D-array with a plus sign.
 3. If they enter '/', fill the 2D-array with a slash.
 4. If they enter '\', fill the 2D-array with a backslash.
 5. If they enter 'p', then print the 2D-array. Note that in the test run I tried to print without first filling the 2D-Array and hence get junk.
 6. If they enter 'q', quit.

Functions:

1. main
2. A function that gets a single character from the user (scans a single letter) and returns it.
3. A different function for each of the filling possibilities (x, +, /, \)
4. A function to print the 2D-array.
5. Feel free to use any more, these are just the ones that I am requiring.

Test Run (input in **bold**):

```
Fill character: *
Empty character: -
What do you want: p
?)@#7?
6GB
r#
#
7bI/
```

Y#Y6W
C/Y#
6WD@7
6W
x#
<Y/Y#

What do you want: +

What do you want: p

-----*-----
-----*-----
-----*-----
-----*-----
-----*-----

-----*-----
-----*-----
-----*-----
-----*-----
-----*-----

What do you want: x

What do you want: p

- *-----* -
--*-----*--
---*---*---
----*-*----
-----*-----
-----*-*-----
----*---*---
--*-----*--
-*-----*--

What do you want: \

What do you want: p

- *-----* -
--*-----*--
---*---*---
----*-*----
-----*-----
-----*-*-----
----*---*---
--*-----*--
-*-----*--

What do you want: /

What do you want: p

-----*
-----*--
-----*--
-----*--

-----*-----
-----*-----
-----*-----
-----*-----
-----*-----
-----*-----
-----*-----
-----*-----

What do you want: **x**

What do you want: **+**

What do you want: **/**

What do you want: **|**

What?

What do you want: ****

What do you want: **p**

*-----
--*-----
---*-----
----*-----
-----*-----
-----*-----
-----*-----
-----*-----
-----*-----
-----*-----
-----*-----
-----*-----
-----*-----
-----*-----
-----*-----

What do you want: **q**

Bye