More fun with pointers

9/18/15
Homework!

• Store the board contents in 2D array
• Complete in versions:
  – Just fill in array and print it
  – Place one R
  – Place one R and one B (using a single function?)
  – Add a loop to place an arbitrary number of pieces
• Remember: You’re not responsible for deciding if someone won
Recall: Pointers

- Variables that store an address in memory
- Get an address with &, use it with *

```c
int* ptr = &x;
int x;
```
int x = 226;
int y = 305;
int* p = &x;
int* q = &y;
p = q;

Which of the following is NOT true after the above?
A. x has value 305
B. y has value 305
C. *p has value 305
D. *q has value 305
E. All of the above are true
int x = 226;
int y = 305;
int* p = &x;
int* q = &y;
p = q;

Which of the following is NOT true after the above?
A. x has value 305
B. y has value 305
C. *p has value 305
D. *q has value 305
E. All of the above are true
Multiple pointers and types

• Using & increases the level of indirection (another * and “pointer to”)

• Using * decreases it

• Example: If p is an int*:
  – *p is an int
  – &p is an int**
Which of the following is a problem with the code below?

```c
int x = 226;       //line 1
int* p = &x;       //line 2
*p = 5;            //line 3
int** q = &p;      //line 4
int* r = *q;       //line 5
```

A. Type error on line 2
B. Type error on line 3
C. Type error on line 4
D. Type error on line 5
E. The code is fine (albeit not useful)
Which of the following is a problem with the code below?

```
int x = 226;        //line 1
int* p = &x;        //line 2
*p = 5;            //line 3
int** q = &p;       //line 4
int* r = *q;        //line 5
```

A. Type error on line 2  
B. Type error on line 3  
C. Type error on line 4  
D. Type error on line 5  
E. The code is fine (albeit not useful)
Mystery functions!