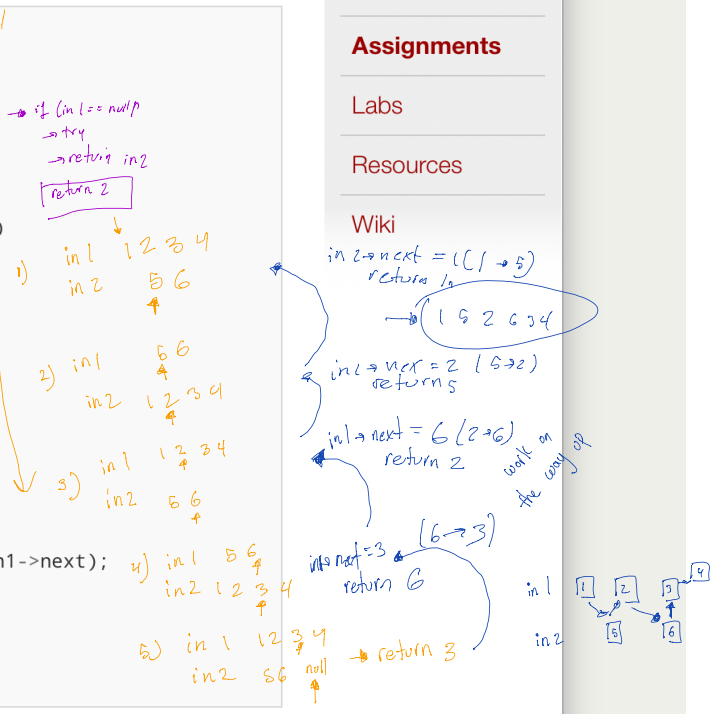
Problem 4 - Linked List Recursion Tracing (13%)

Consider the following C++ code. For the following questions, all of the points for this problem will be assigned based on your explanation, since we have full faith in your ability to run this program and copy down the answer.

To show work, you can draw a call tree or box diagram of the function calls using some simplified substitution of your choice rather than pointer values (e.g. "p3" for a pointer to a node with value 3). Submit your answers as a PDF (using some kind of illustration software or scanned handwritten notes where you use your phone to convert to PDF) showing your work and derivations supporting your final answer. **You must name the file** `q4_answers.pdf`.

```
struct Node {
    int val;
    Node*  next;
};

Node* llrec(Node* in1, Node* in2)
{
    if(in1 == nullptr) {
        return in2;
    }
    else if(in2 == nullptr) {
        return in1;
    }
    else {
        in1->next = llrec(in2, in1->next);
        return in1;
    }
}
```



Question a: What linked list is returned if `llrec` is called with the input linked lists **in1** = `1,2,3,4` and **in2** = `5,6`? returns 1 5 2 6 3 4

Question b: What linked list is return if `llrec` is called with the input linked lists **in1** = `nullptr` and **in2** = `2`? will return 2

Problem 5 - NOT GRADED (PRACTICE