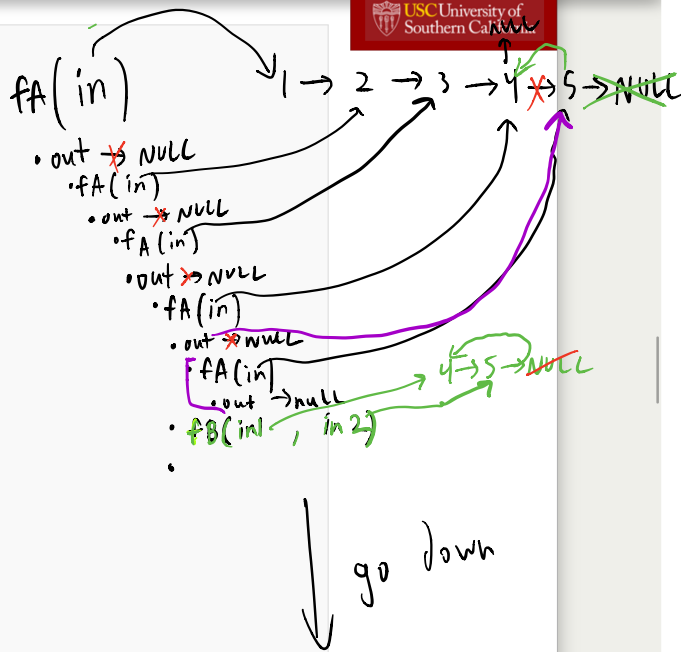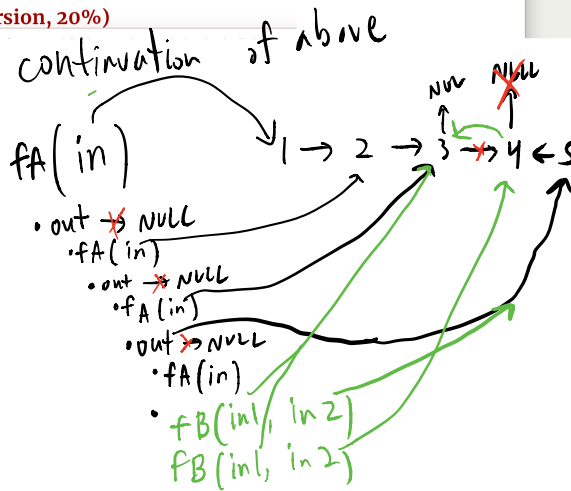answer.

```cpp
struct Node {
    int value;
    Node *next;
};

void funcB (Node* in1, Node* in2);

Node* funcA (Node* in)
{
    if (in == NULL) return NULL;
    Node* out = NULL;
    if (in->next != NULL)
    {
        out = funcA (in->next);
        funcB (in, out);
        in->next = NULL;
        return out;
    }
    return in;
}

void funcB (Node* in1, Node* in2)
{
    if (in2->next != NULL)
    {
        funcB (in1, in2->next);
        return;
    }
    in2->next = in1;
}
```
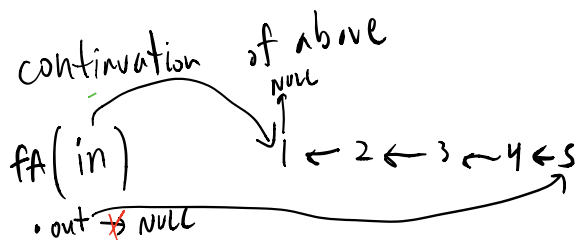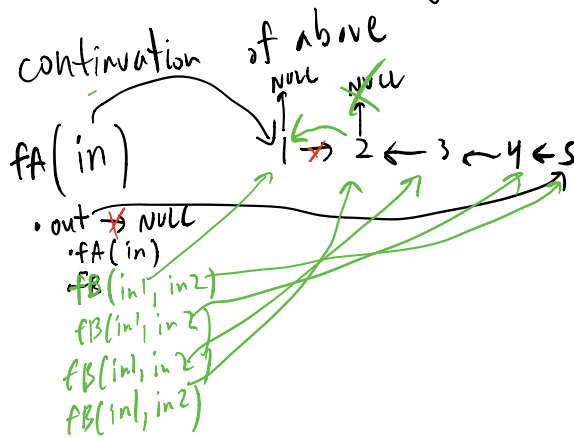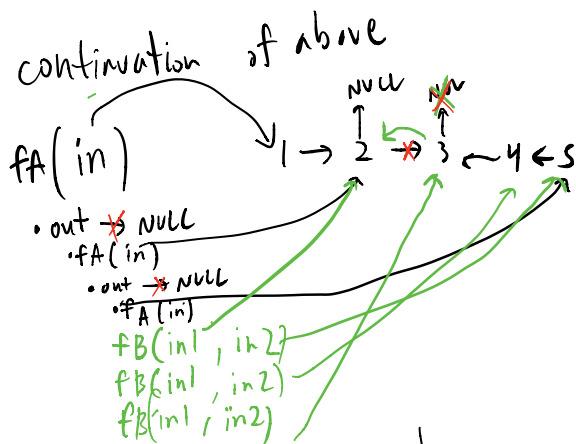
*Handwritten annotations:*

fA(in)    1 → 2 → 3 → 4 ⤫ 5 → NULL

• out ⤫ NULL
 • fA(in)
• out ⤫ NULL
 • fA(in)
• out ⤫ NULL
 • fA(in)
• out ⤫ NULL
 • fA(in)
 • out → null
• fB(in1, in2)    4 → 5 → NULL

go down ↓

## Problem 6 (Linked Lists, Recursion, 20%)

continuation of above

fA(in)    1 → 2 → 3 ⤫ 4 ← 5    NULL NULL

• out ⤫ NULL
 • fA(in)
• out ⤫ NULL
 • fA(in)
• out ⤫ NULL
 • fA(in)
•
 fB(in1, in2)
 fB(in1, in2)

continuation of above

fA ( in )

NULL

1 → 2 →✗ 3 ⟿ 4 ⟵ 5

• out ✗ NULL
    • fA ( in )
        • out ✗ NULL
            • fA ( in )
fB ( in1 , in2 )
fB ( in1 , in2 )
fB ( in1 , in2 )

↓

continuation of above

fA ( in )

NULL        NULL

1 ✗→ 2 ⟵ 3 ⟿ 4 ⟵ 5

• out ✗ NULL
    • fA ( in )
• fB ( in1 , in2 )
    fB ( in1 , in2 )
    fB ( in1 , in2 )
    fB ( in1 , in2 )

•

↓

continuation of above

fA ( in )

NULL

1 ⟵ 2 ⟵ 3 ⟿ 4 ⟵ 5

• out ✗ NULL

Result is a Nodepointer (*out) that points to reversed
            linked list.

Node*  out → [5] → [4] → [3] → [2] → [1] → NULL