I’ll trace the 2 functions with an example number. I will take the user entered number as 234;

Let’s trace the function mystery1(int);

mystery1(234)

When n = 0 since the n !> 0 the if condition get false and function will end.

printf(2)

mystery1(0)

Like in the previous step

printf(3)

mystery1(2)

Since n > 0;

Since the first line is a print function it will print 234%10 which is 4. Then mystery1 functions is called with number 234/10 which is 23.

printf(4)

myster1(23)

So at the end output number which has printed will be 432 which is the reverse of user input number.

Now let’ trace mystery2(int ) function. In here unlike in above function the printf() function is below the recursive call.

//First print line

printf(4)

234%10

//Second print line

printf(2)

n !> 0

mystery2(0)---2/10

n > 0

mystery2(2)----23/10

n > 0

mystery2(23)---234/10

n > 0

mystery2(234)

//Third print line

printf(3)

23%10

2%10

In here since the print statement is after the recursive function it will work after running through the recursion. Then the first print line will be associated with the last value obtained by the recursive function.

At the end since the condition will get false the function will end without a print statement.

So the output number will be the same number which is 234.

The above explanation is valid for all valid arguments for the two recursive functions.