

2. Problems on Recursion

Basic Recursion Problem

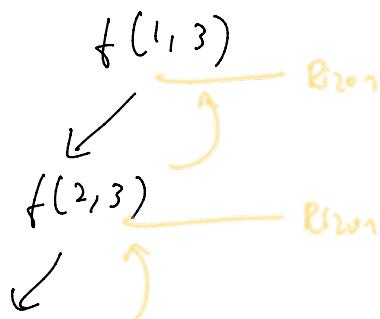
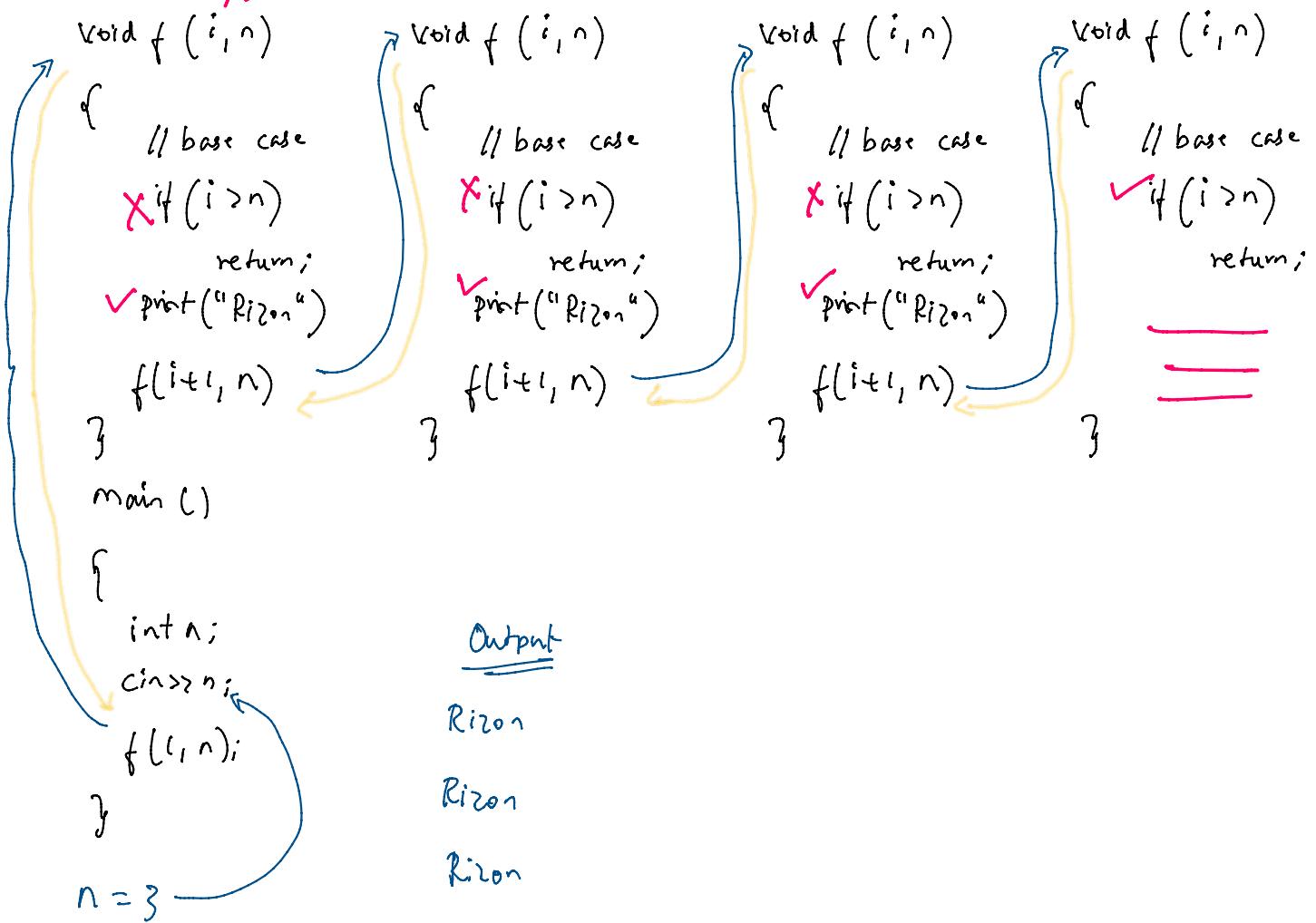
① Print Name n times.

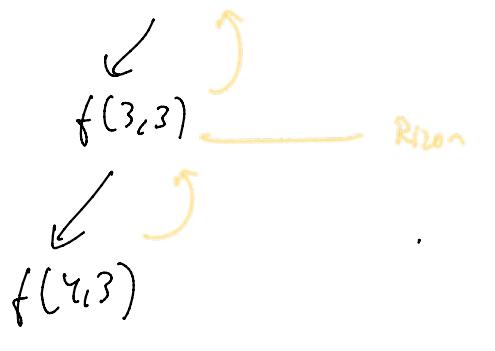
1, 3

2, 3

3, 3

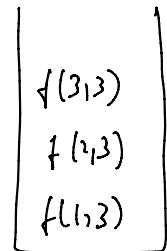
4, 3





L.C $\Rightarrow O(n)$

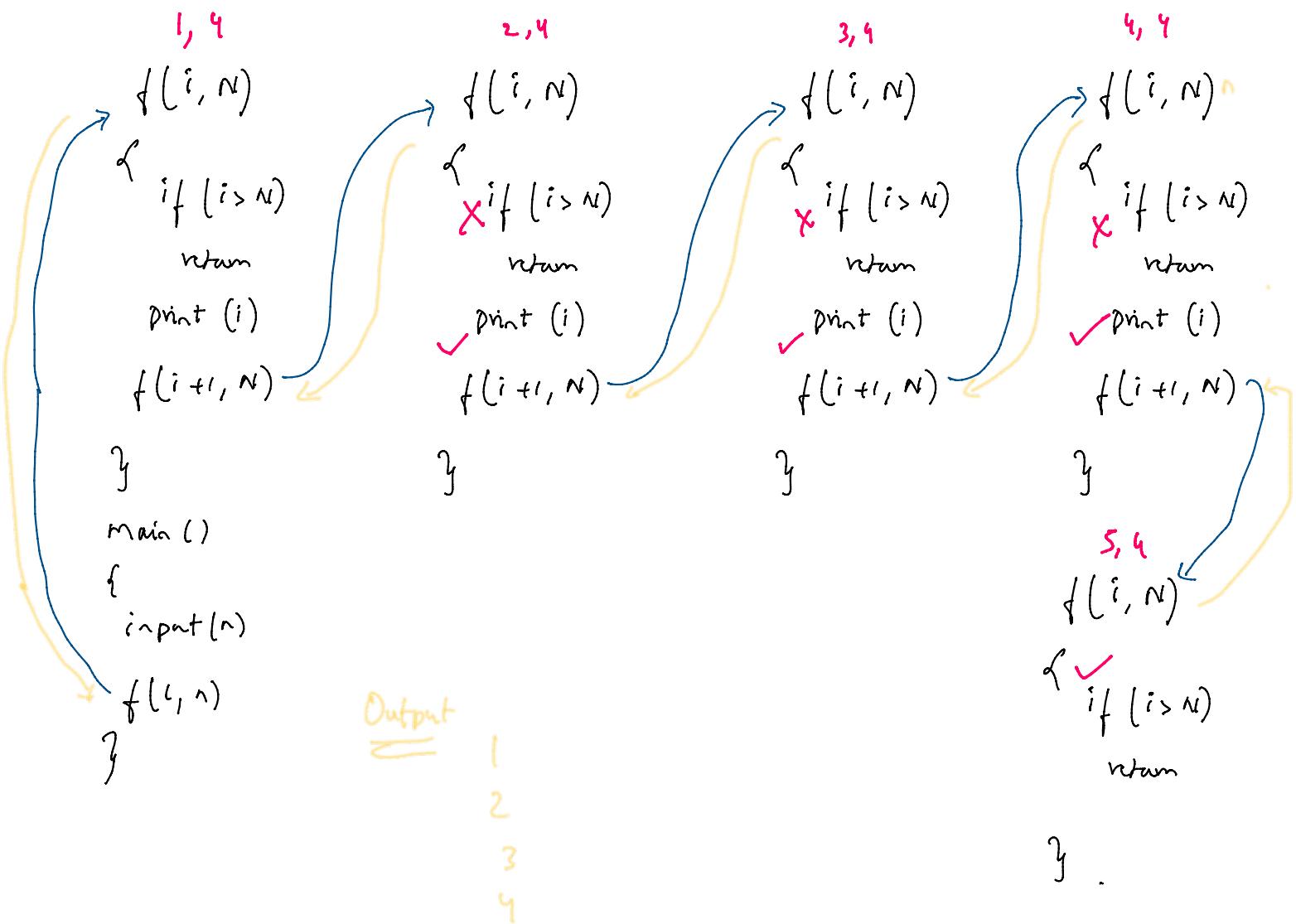
S.C $\Rightarrow O(n)$ (Stack Space)



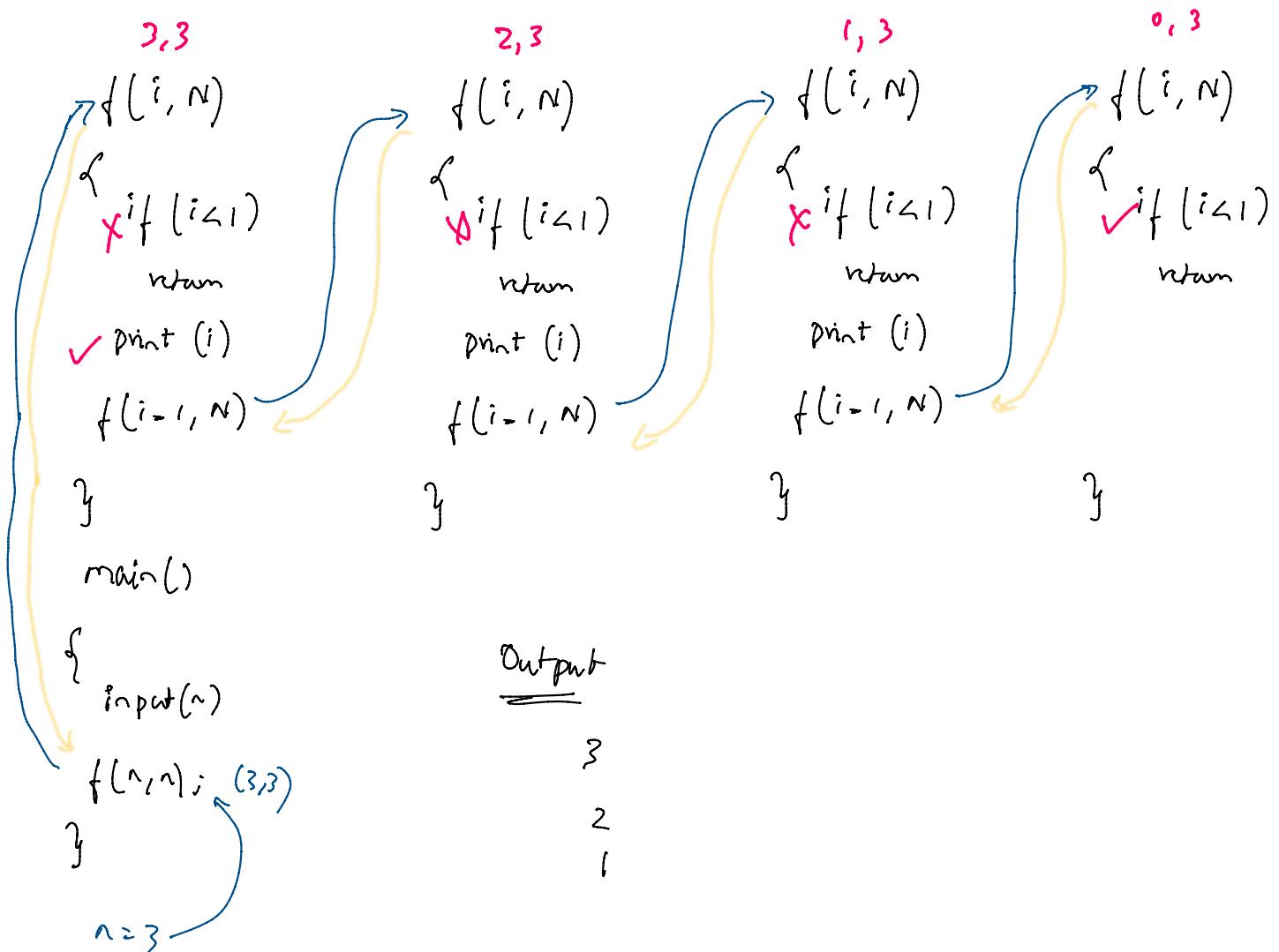
Stack

(2) Print linearly from 1 to N

$N=4$ Output $\Rightarrow 1 \ 2 \ 3 \ 4$

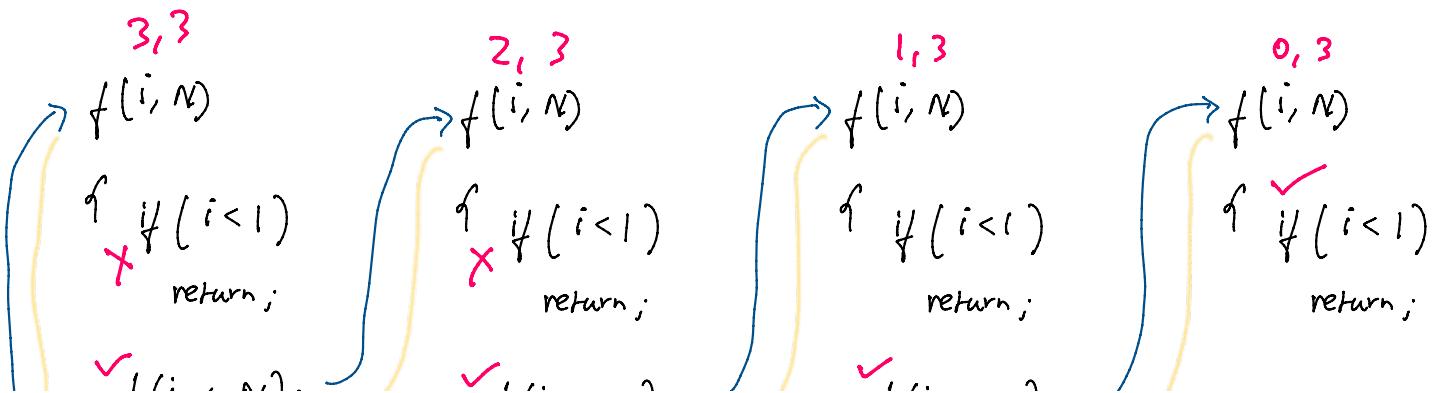


Q Print $N \rightarrow 1$ $N=3 \Rightarrow 3 \ 2 \ 1$



Backtracking

Q: Print from 1 to N but without using '+1' $\Rightarrow f(i+1, N)$.



```

    return;
    f(i-1, N);
    print(i);
}
main()
{
    input(N)
    f(N, N);
}
n = 3

```

return;
return;
return;

} } }

return;

} }

Your printing it after all the function
call get over. (Opposit side we did)

Output

1

2

3

Q: Print from N to 1 without using ' -1 ' $f(i-1, N) \times$

```

1,3
f(i,n)
{
if(i >= n)
    return
f(i+1, n)
print(i) //1
}

2,3
f(i,n)
{
if(i >= n)
    return
f(i+1, n)
print(i) //2
}

3,3
f(i,n)
{
if(i >= n)
    return
f(i+1, n)
print(i) //3
}

4,3
f(i,n)
{
if(i >= n)
    return
}

```

scanf("%d", &i);

printf("%d\n", i);

scanf("%d", &n);

main() {

 input(n)

 f(1, n)

}

n = 3

Output :

3

2

1