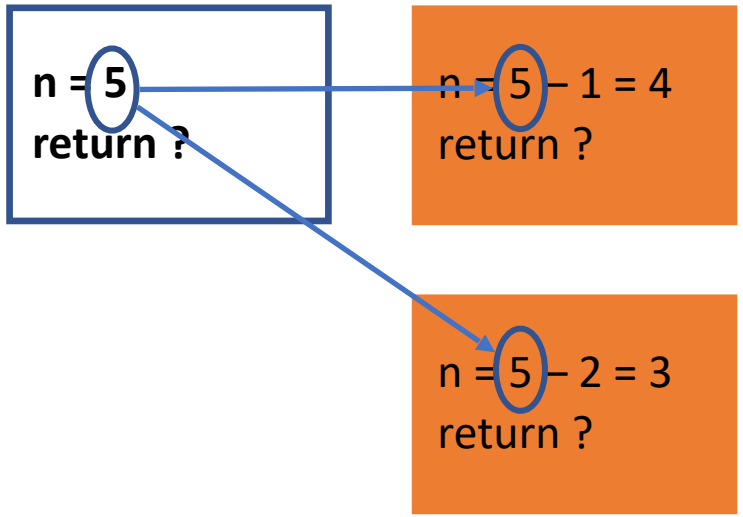
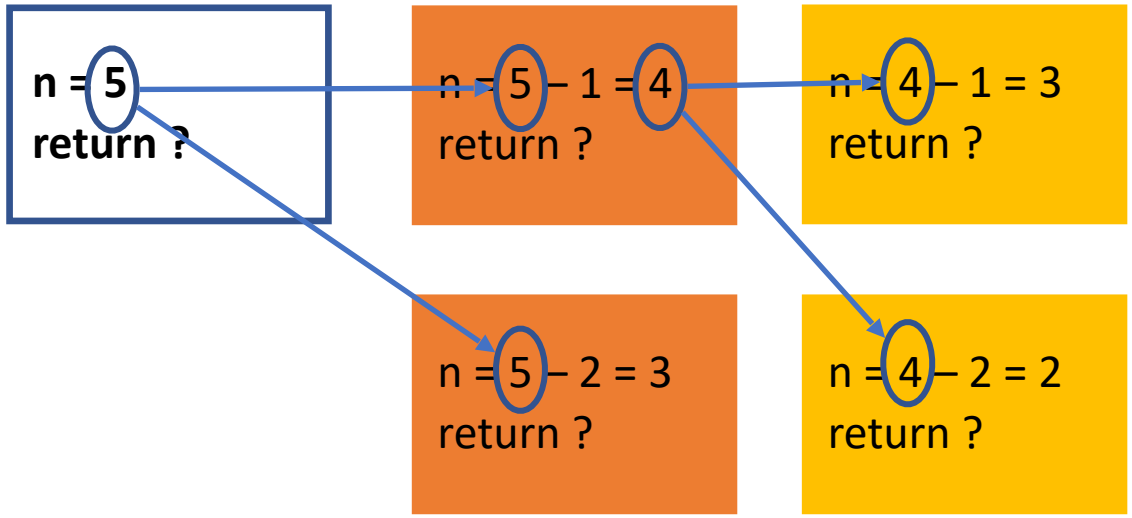
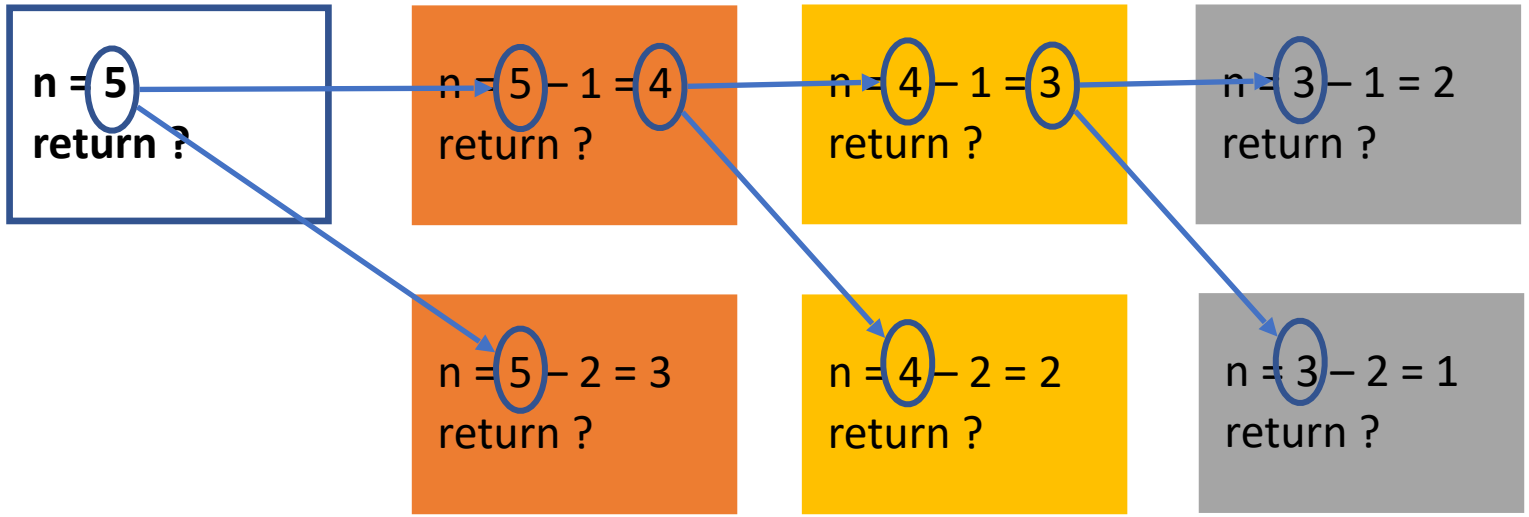
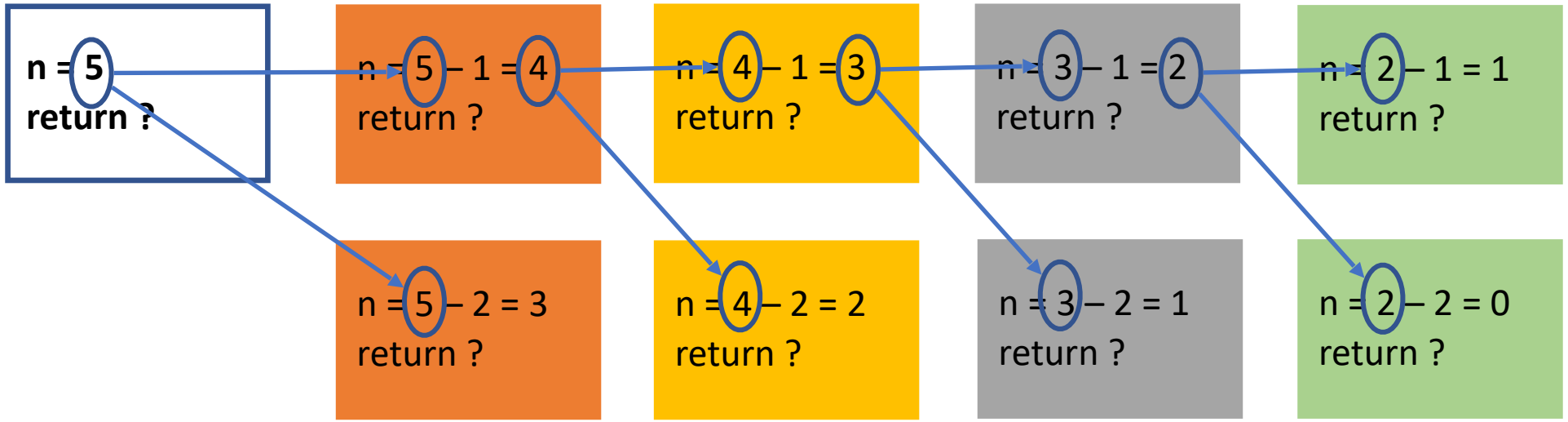
```
public static long fibonacci(long n)
  {
    if (n == 0)
      return 1;
    else if (n == 1)
      return 1;
    else
      return fibonacci(n - 1) + fibonacci(n - 2);
  }
```

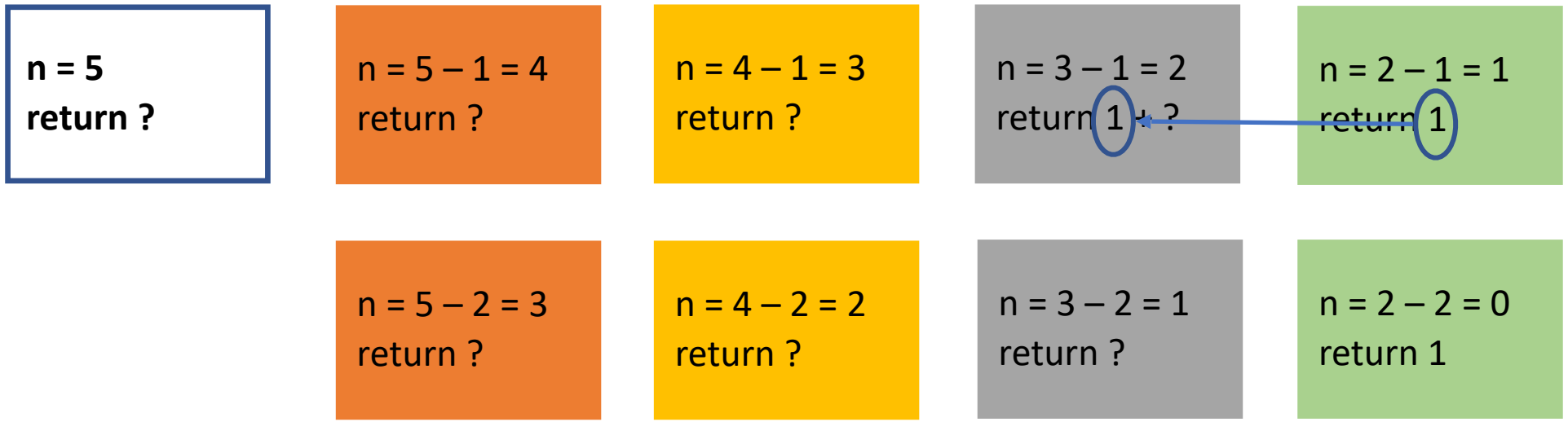
n = 5
return ?

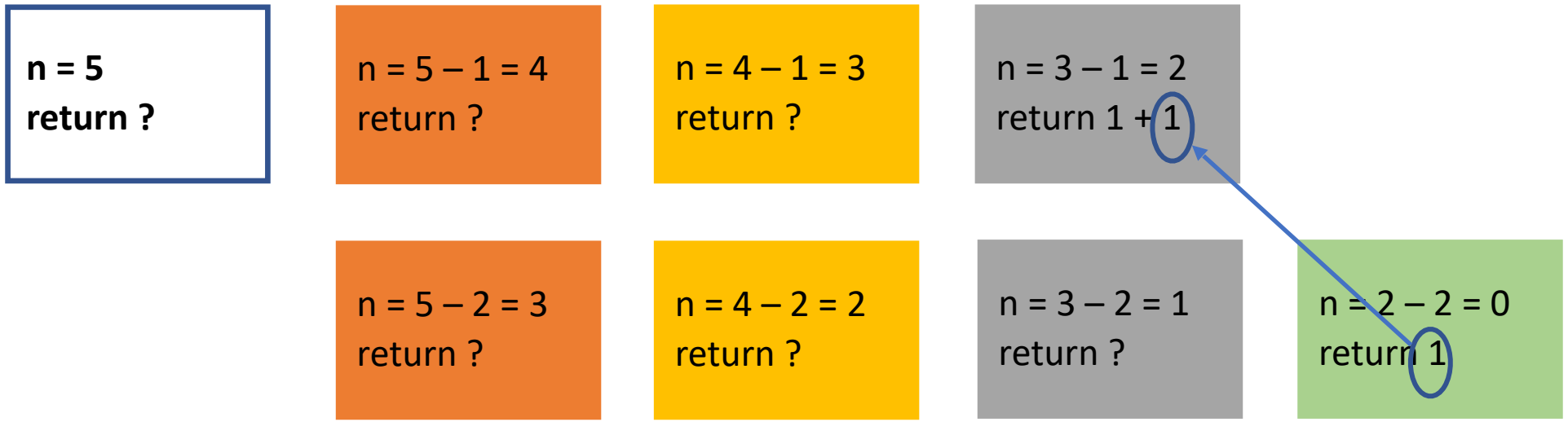

n = 5
return ?
n = 5 – 1 = 4
return ?
n = 5 – 2 = 3
return ?

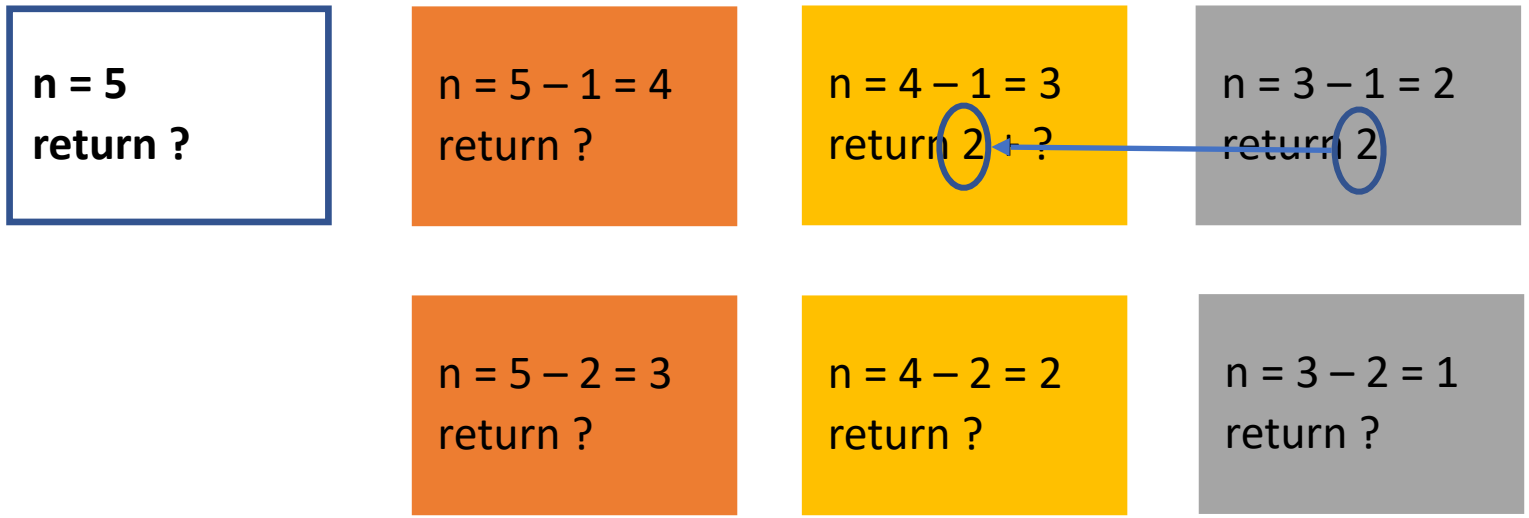

n = 5
return ?
n = 5 – 1 = 4
return ?
n = 4 – 1 = 3
return ?
n = 5 – 2 = 3
return ?
n = 4 – 2 = 2
return ?

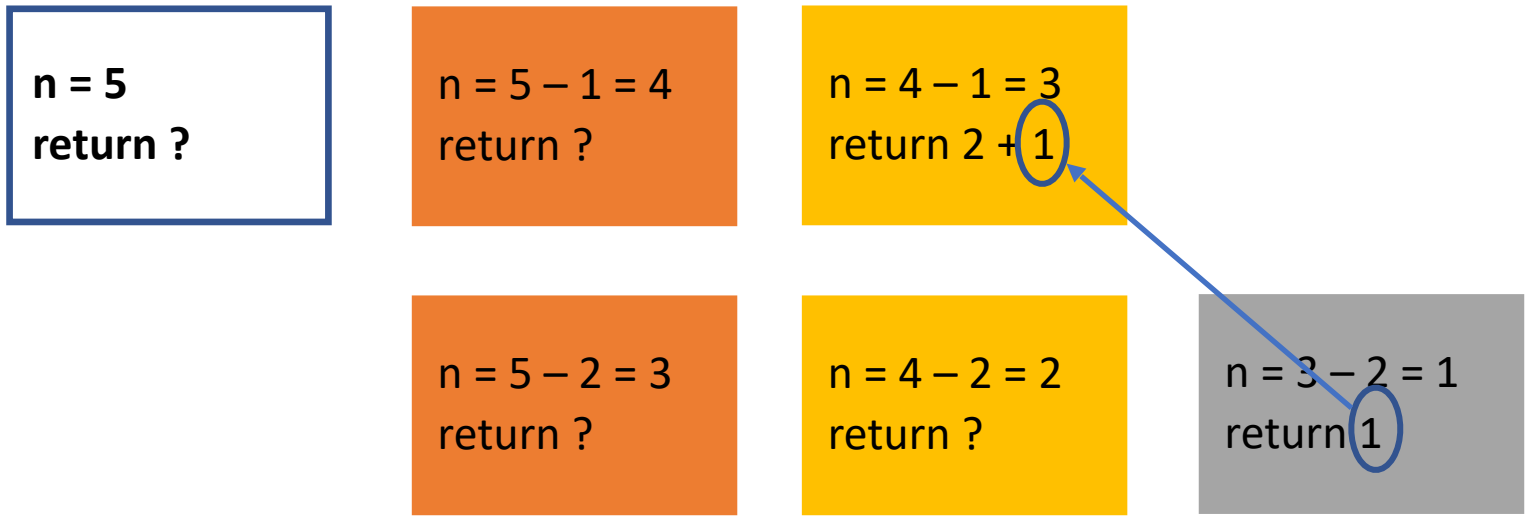

n = 5
return ?
n = 5 – 1 = 4
return ?
n = 4 – 1 = 3
return ?
n = 3 – 1 = 2
return ?
n = 5 – 2 = 3
return ?
n = 4 – 2 = 2
return ?
n = 3 – 2 = 1
return ?

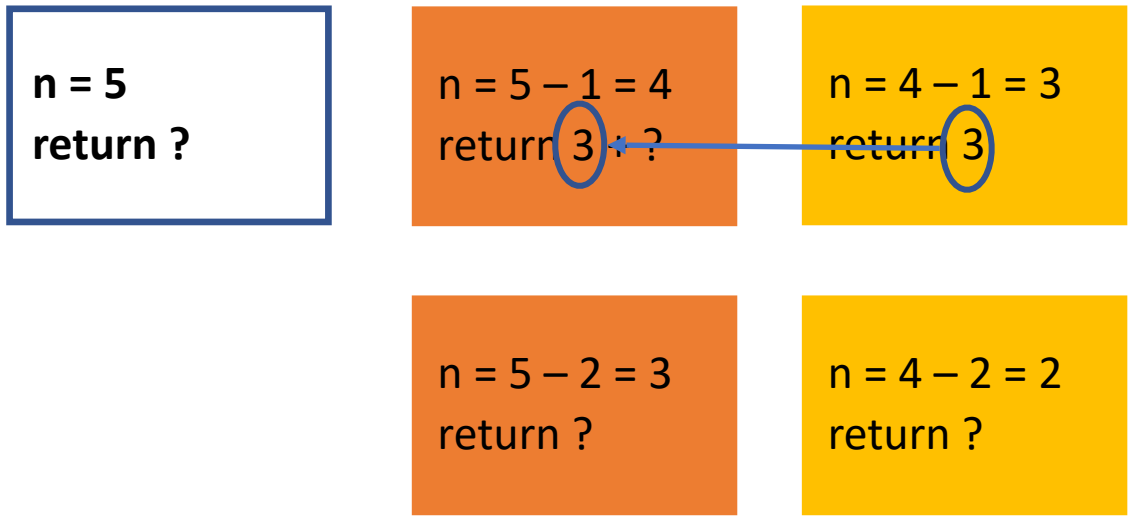

n = 5
return ?
n = 5 – 1 = 4
return ?
n = 4 – 1 = 3
return ?
n = 3 – 1 = 2
return ?
n = 2 – 1 = 1
return ?
n = 5 – 2 = 3
return ?
n = 4 – 2 = 2
return ?
n = 3 – 2 = 1
return ?
n = 2 – 2 = 0
return ?

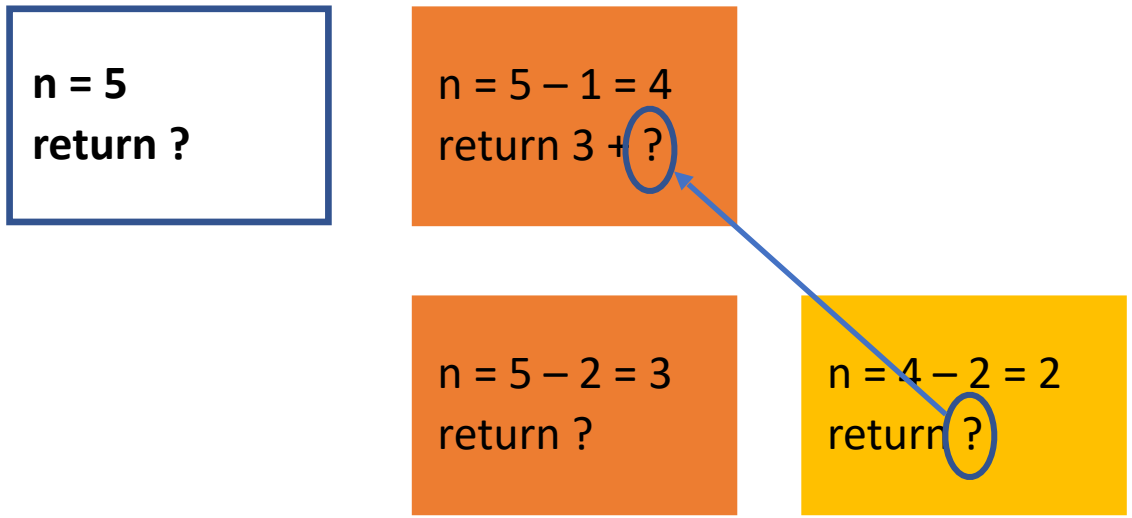

n = 5
return ?
n = 5 – 1 = 4
return ?
n = 4 – 1 = 3
return ?
n = 3 – 1 = 2
return 1 + ?
n = 2 – 1 = 1
return 1
n = 5 – 2 = 3
return ?
n = 4 – 2 = 2
return ?
n = 3 – 2 = 1
return ?
n = 2 – 2 = 0
return 1

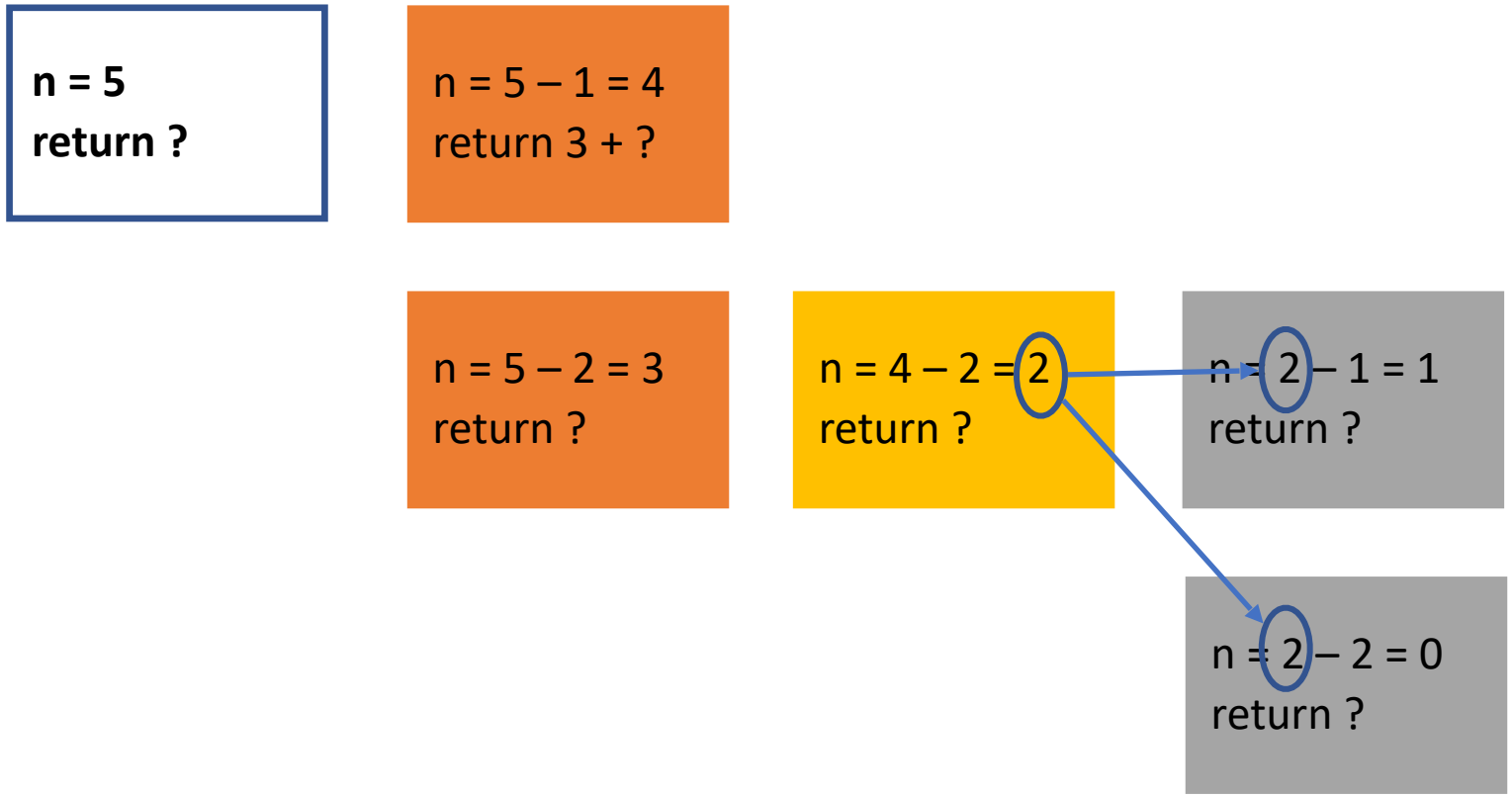

n = 5
return ?
n = 5 – 1 = 4
return ?
n = 4 – 1 = 3
return ?
n = 3 – 1 = 2
return 1 + 1
n = 5 – 2 = 3
return ?
n = 4 – 2 = 2
return ?
n = 3 – 2 = 1
return ?
n = 2 – 2 = 0
return 1

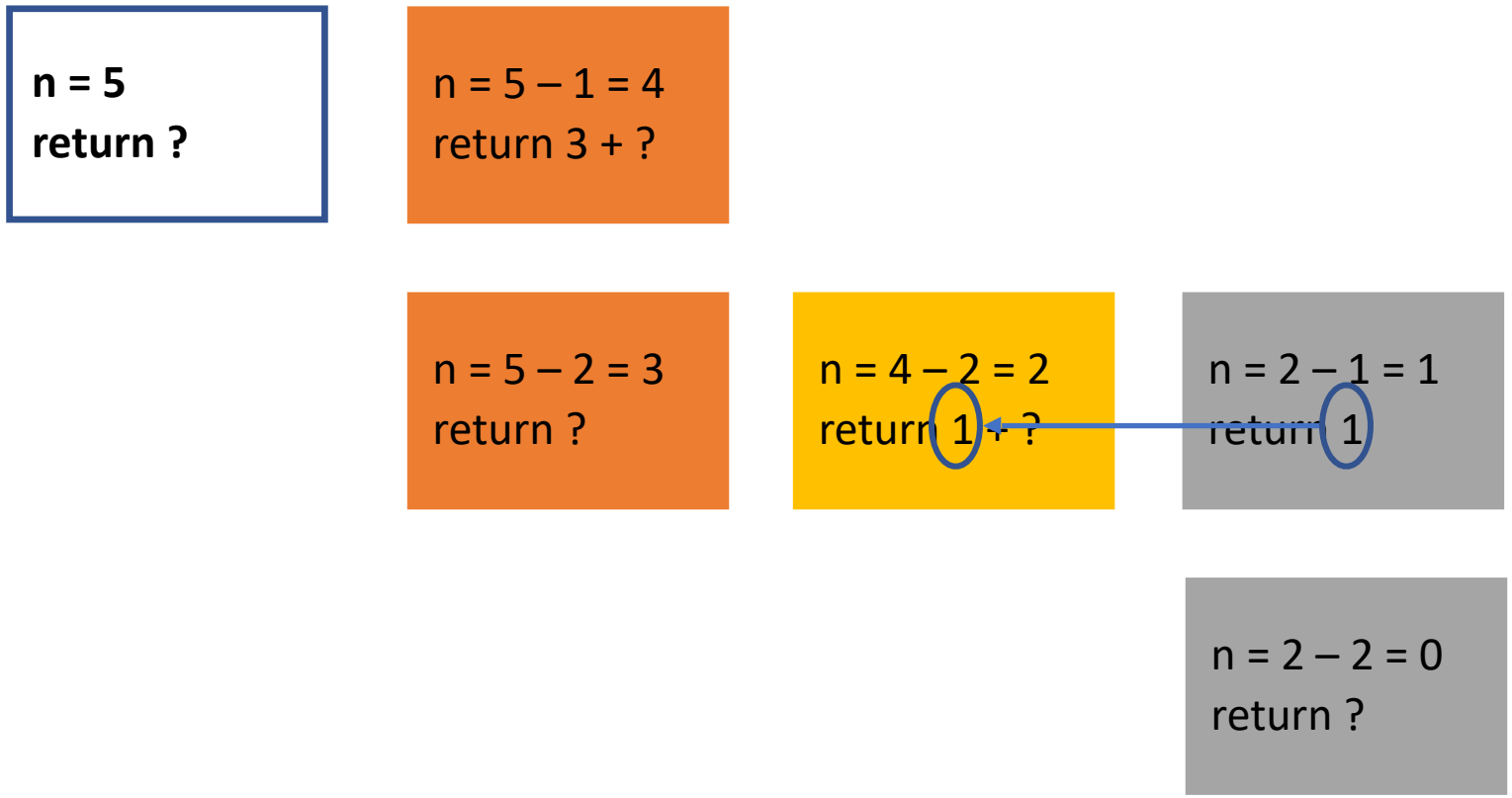

n = 5
return ?
n = 5 – 1 = 4
return ?
n = 4 – 1 = 3
return 2 + ?
n = 3 – 1 = 2
return 2
n = 5 – 2 = 3
return ?
n = 4 – 2 = 2
return ?
n = 3 – 2 = 1
return ?

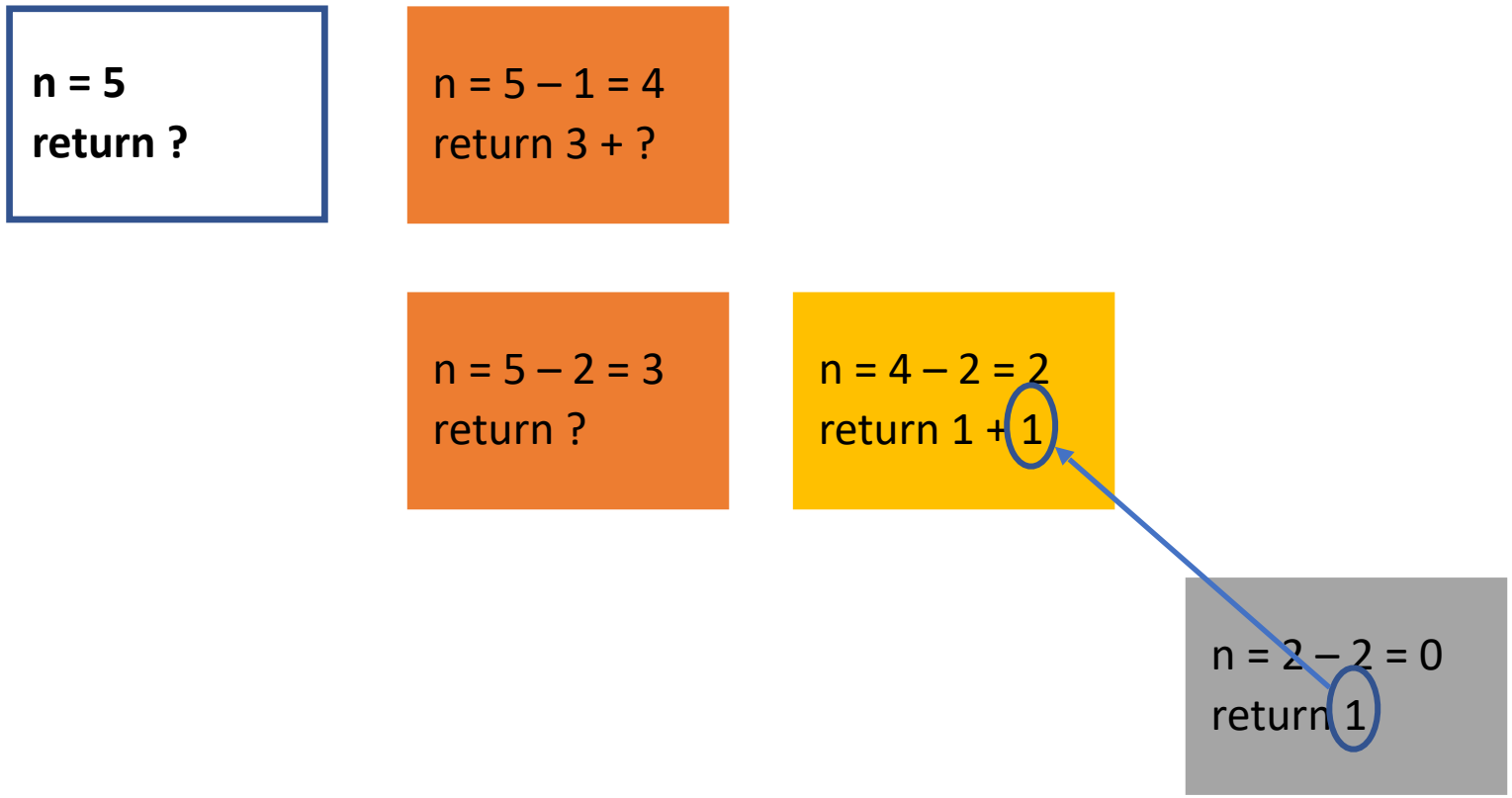

n = 5
return ?
n = 5 – 1 = 4
return ?
n = 4 – 1 = 3
return 2 + 1
n = 5 – 2 = 3
return ?
n = 4 – 2 = 2
return ?
n = 3 – 2 = 1
return 1

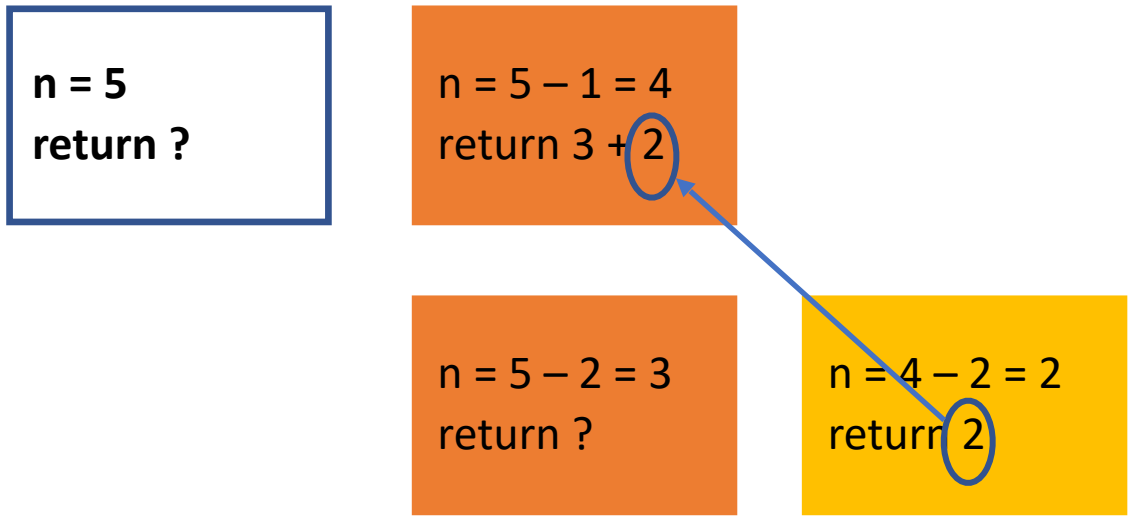

n = 5
return ?
n = 5 – 1 = 4
return 3 + ?
n = 4 – 1 = 3
return 3
n = 5 – 2 = 3
return ?
n = 4 – 2 = 2
return ?

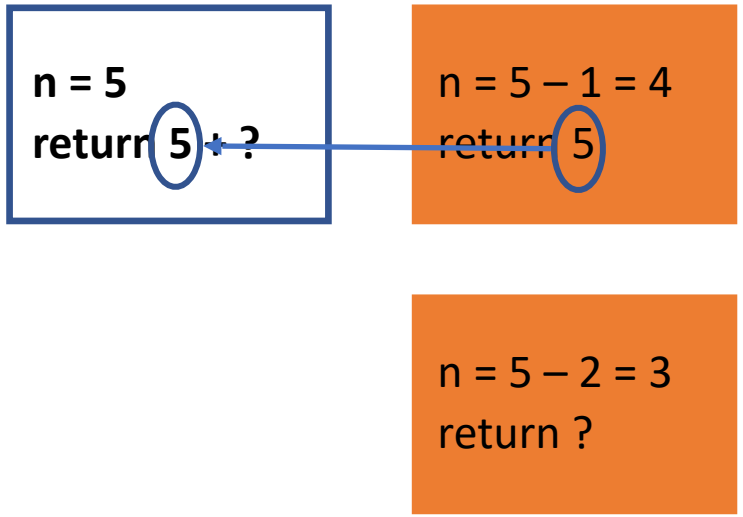

n = 5
return ?
n = 5 – 1 = 4
return 3 + ?
n = 5 – 2 = 3
return ?
n = 4 – 2 = 2
return ?

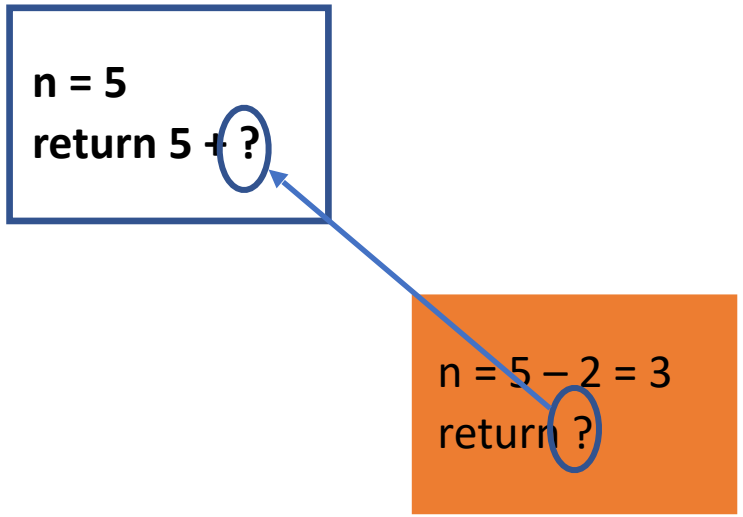

n = 5
return ?
n = 5 – 1 = 4
return 3 + ?
n = 5 – 2 = 3
return ?
n = 4 – 2 = 2
return ?
n = 2 – 1 = 1
return ?
n = 2 – 2 = 0
return ?


n = 5
return ?
n = 5 – 1 = 4
return 3 + ?
n = 5 – 2 = 3
return ?
n = 4 – 2 = 2
return 1 + ?
n = 2 – 1 = 1
return 1
n = 2 – 2 = 0
return ?

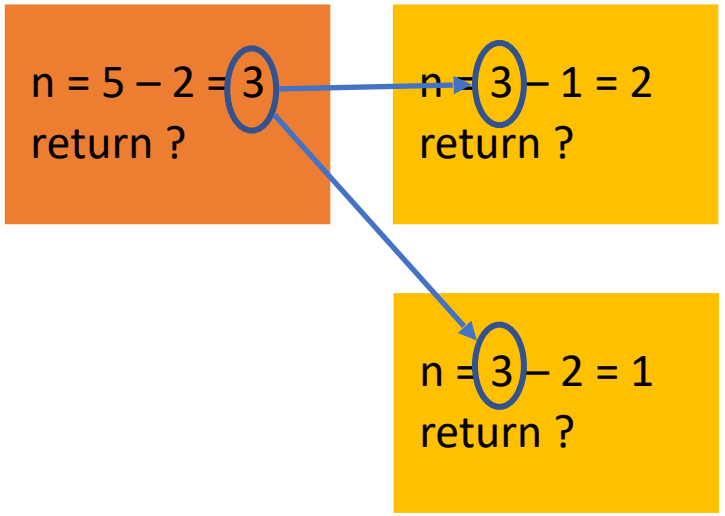

n = 5
return ?
n = 5 – 1 = 4
return 3 + ?
n = 5 – 2 = 3
return ?
n = 4 – 2 = 2
return 1 + 1
n = 2 – 2 = 0
return 1


n = 5
return ?
n = 5 – 1 = 4
return 3 + 2
n = 5 – 2 = 3
return ?
n = 4 – 2 = 2
return 2

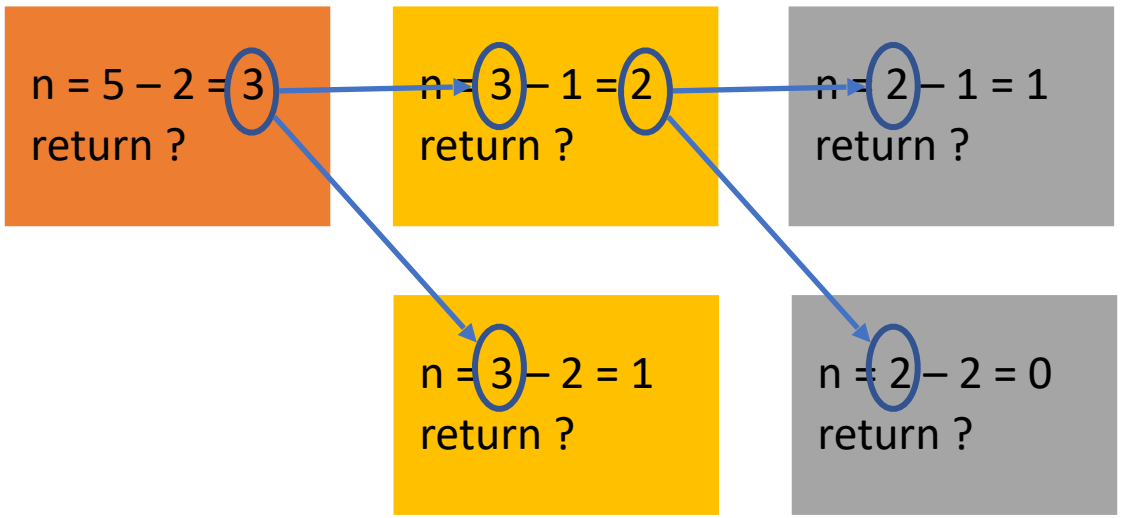

n = 5
return 5 + ?
n = 5 – 1 = 4
return 5
n = 5 – 2 = 3
return ?


n = 5
return 5 + ?
n = 5 – 2 = 3
return ?

n = 5
return 5 + ?



n = 5
return 5 + ?


n = 5 – 2 = 3
return ?
n = 3 – 1 = 2
return ?
n = 2 – 1 = 1
return ?
n = 3 – 2 = 1
return ?
n = 2 – 2 = 0
return ?

n = 5
return 5 + ?

n = 5 – 2 = 3
return ?

n = 3 – 1 = 2
return 1 + ?

n = 2 – 1 = 1
return 1

n = 3 – 2 = 1
return ?

n = 2 – 2 = 0
return ?

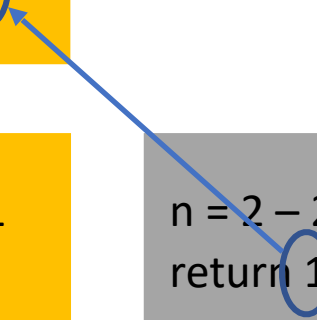
n = 5
return 5 + ?

n = 5 – 2 = 3
return ?

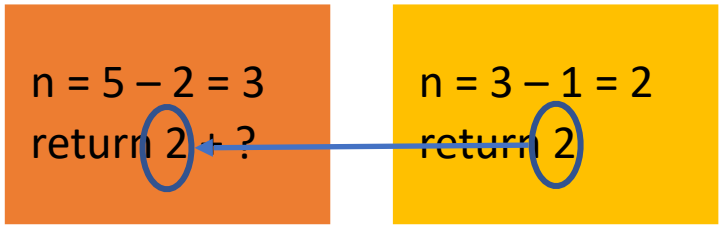
n = 3 – 1 = 2
return 1 + 1

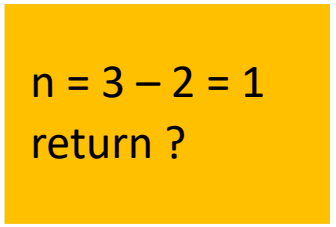
n = 3 – 2 = 1
return ?

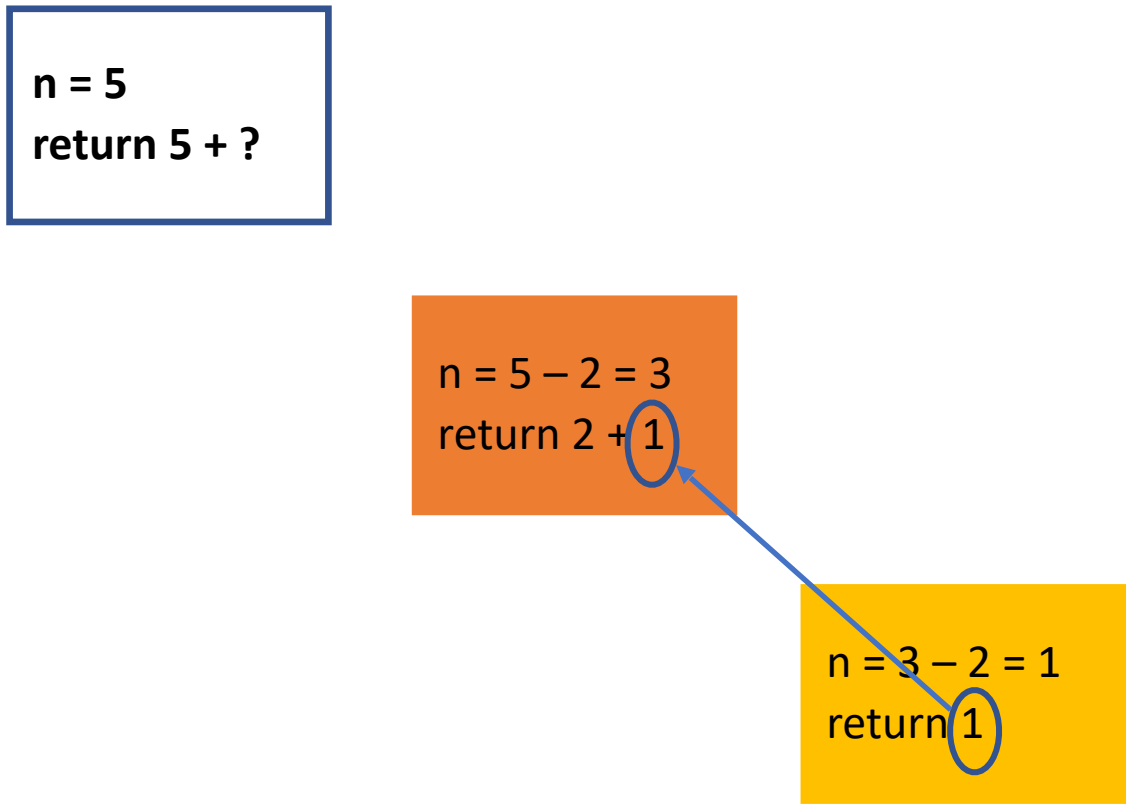
n = 2 – 2 = 0
return 1

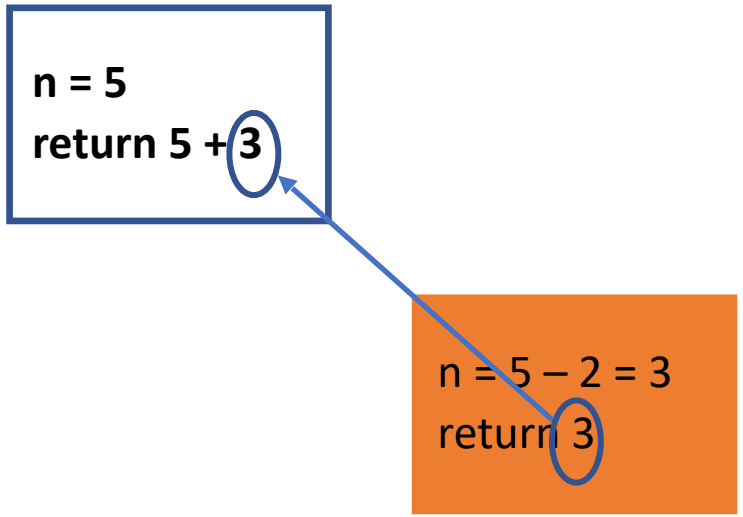


n = 5
return 5 + ?






n = 5
return 5 + ?
n = 5 – 2 = 3
return 2 + 1
n = 3 – 2 = 1
return 1


n = 5
return 5 + 3
n = 5 – 2 = 3
return 3

n = 5
return 8

Results

fibonacci(5) → 1

fibonacci(4) → 1

fibonacci(3) → 2

fibonacci(2) → 3

fibonacci(1) → 5

fibonacci(0) → 3