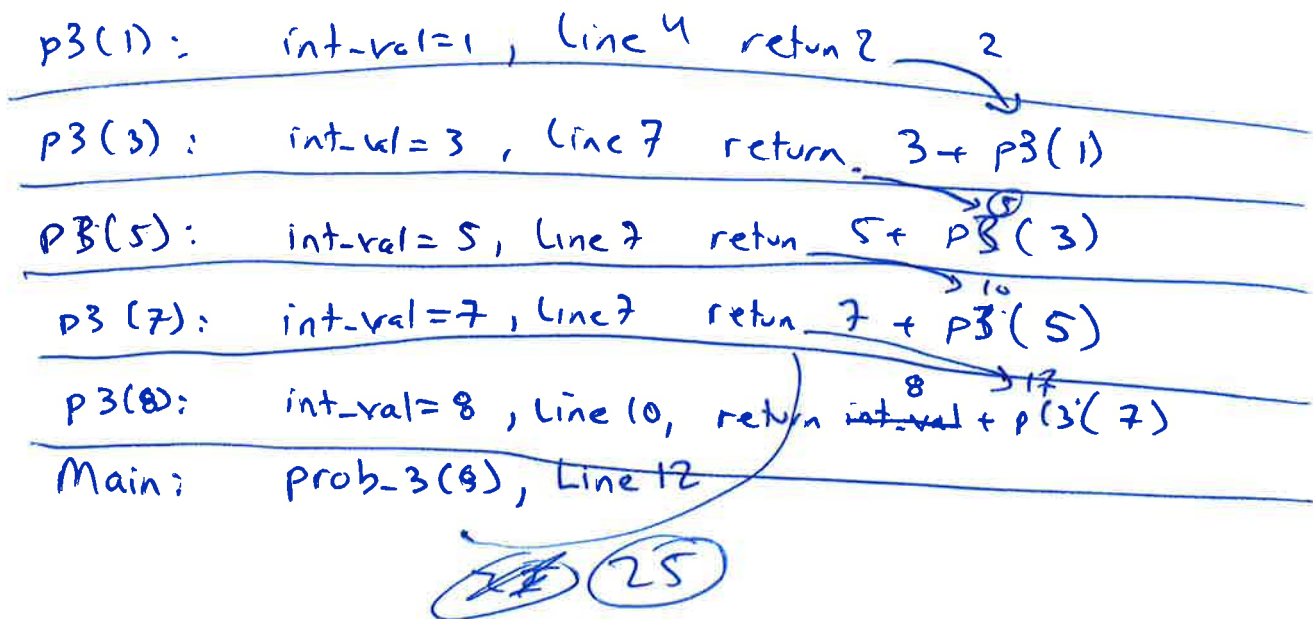


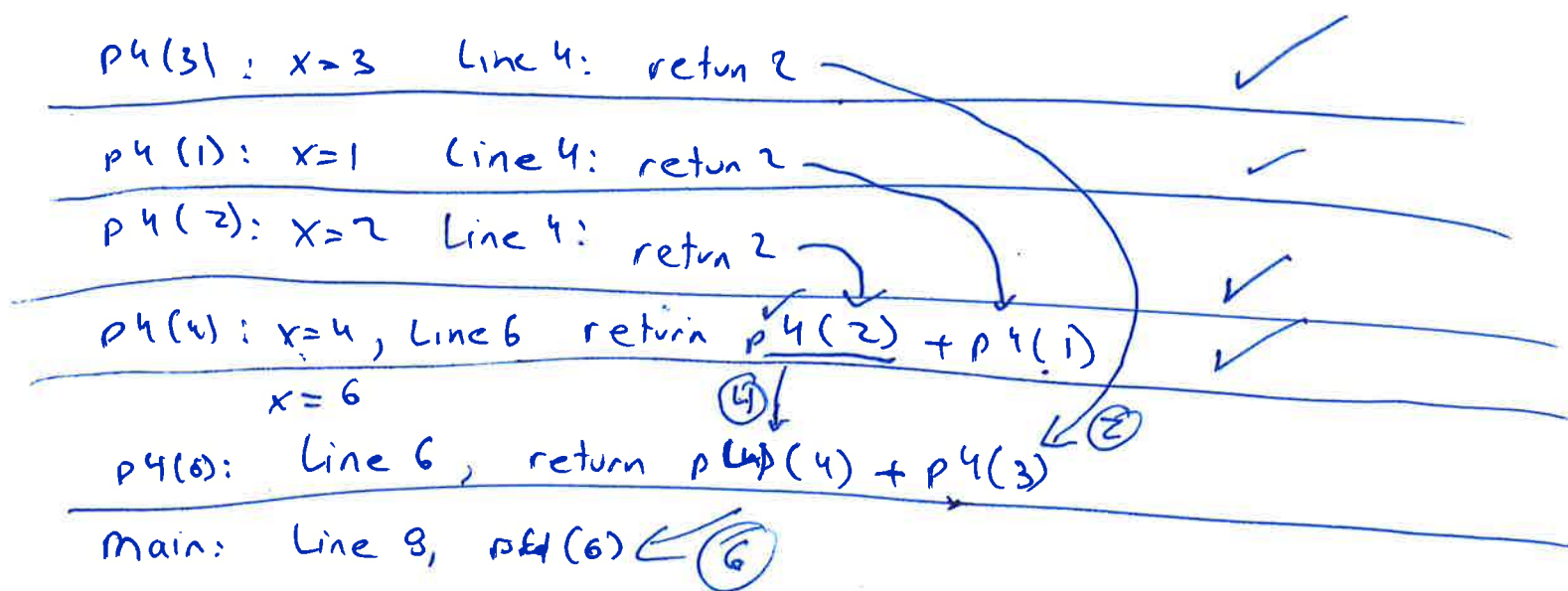
Given the following code, draw the recursive trace in memory and indicate what the final result will be for the function call below:

```
1 def prob_3( int_val ):
2     if int_val <= 1:
3         return 2
4
5     if int_val % 2 == 1:
6         return int_val + prob_3( int_val - 2 )
7
8     else:
9         return int_val + prob_3( int_val - 1 )
10
11
12 prob_3( 8 ) ←
```



Given the following code, draw the recursive trace in memory and indicate what the final result will be for the function call below:

```
def prob_4( x ):  
    if x <= 3:  
        return 2  
  
    return prob_4( x - 2 ) + prob_4( x - 3 )  
  
prob_4(6)
```



Given the following code, draw the recursive trace in memory and indicate what the final result will be for the function call below:

```
1 def foo(x):
2     if x <= 0:
3         return 5
4
5     if x % 2 == 0:
6         return x + bar(x-1)
7
8     elif x % 2 == 1:
9         return x + foo(x-3)
10
11
12
13 def bar(x):
14
15     if x <= 2:
16         return 4
17
18     if x % 2 == 0:
19         return x + foo(x-3)
20
21     elif x % 2 == 1:
22         return x + bar(x-1)
23
24
25 foo(12)
```

Handwritten recursive trace:

- $b(2): x=2$ Line 16 return 4 ✓
- $b(3): x=3$ Line 22 return $3 + b(2)$ ✓
- $f(4): x=4$, Line 7, return $4 + b(3)$ ✓
- $f(7): x=7$, Line 10, return $7 + f(4)$ ✓
- $b(10): x=10$, Line 19, return $10 + f(7)$ ✓
- $b(11): x=11$, Line 27: return $11 + b(10)$ ✓
- $f(12): x=12$, Line 7: return $12 + bar(11)$ ✓
- main: $foo(12)$

Final result: 51