Recursion 1

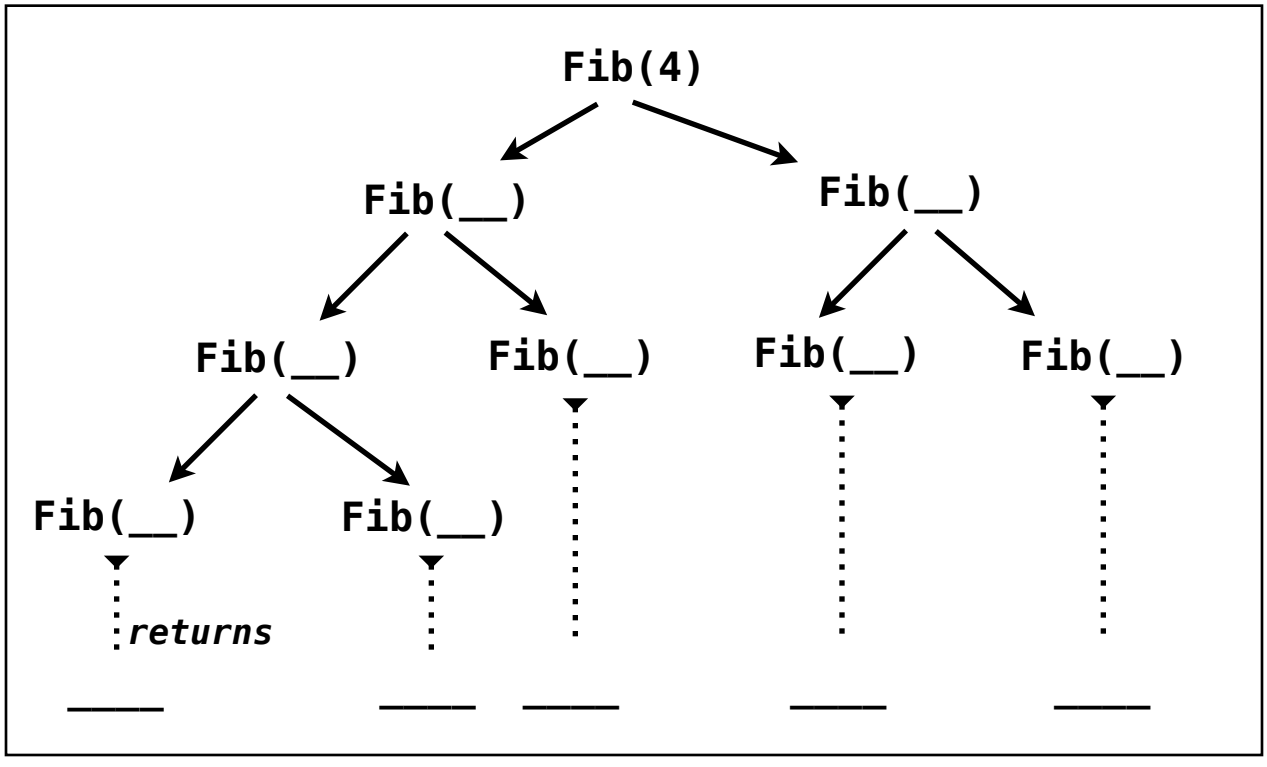
Fibonacci

(a) Fill in the blanks to complete the Fibonacci function.

```
Fibonacci(n)

if  _____________________________ :

    report ______

if _____________________________ :

    report ______

else:

    report ____________________________________________
```

(b) Circle and label the *base case* and the *recursive case* in the code above.

(c) Fill in the recursive call tree below that represents the call: `Fib(4)`.



Palindrome

(a) A palindrome is a word that is spelled the same way forwards and backwards. In other words, the first letter must equal the last letter, the second letter must equal the second to last letter … etc.

Using the above information, fill in the recursive palindrome function. You have access to these two functions:

`all-but-first-letter-of( word )`
reports *word* with the first letter gone.

`all-but-last-letter-of( word )`
reports *word* with the last letter gone.

```
is (word) a palindrome?

if  ____________________________________________ :

    report ______

else:

    if ____________________________________________:

        report ____________________________________

    else:

        report ______
```

(b) Fill in the progression of calls to: is (racecar) a palindrome?