```
FIB(n):
1  if n < 0:
2      return null
3  if n = 0 or n = 1:                  // you can also
4      return n                        // add comments!
5  return FIB(n − 1) + FIB(n − 2)
```

```
if n < 0:
    return null
if n = 0 or n = 1:
    return n

let x ← 0
let y ← 1
for i ← 2 to n − 1:    // so dynamic!
    let z ← x + y
    x ← y
    y ← z

return x + y
```

```
FIB (n):
if n < 0:
    return null
if n = 0 or n = 1:
    return n

let x ← 0
▷ for instance
                                ▷ for instance
for
for                             ▷ used #no-keyword[for]
superloop                       ▷ used #keyword[superloop]
□                               ▷ used #box(..., keyword[superloop])
let y ← 1
for i ← 2 to n − 1:             ▷ so dynamic!
    let z ← x + y if x ≠ 0
    x ← y
    y ← z


return x + y
```

```
def fib(n):
  if n < 0:
    return None
  if n == 0 or n == 1:        # this comment is
    return n                  # normal raw text
  return fib(n-1) + fib(n-2)
```

```
def fib(n):
    if n < 0:
        return None
    if n == 0 or n == 1:        # this comment is
        return n                # normal raw text
    return fib(n-1) + fib(n-2)
```