

RETURN AND PRINT

An Introduction to Computer Science



Returning and Printing

- Printing: Putting things on the console

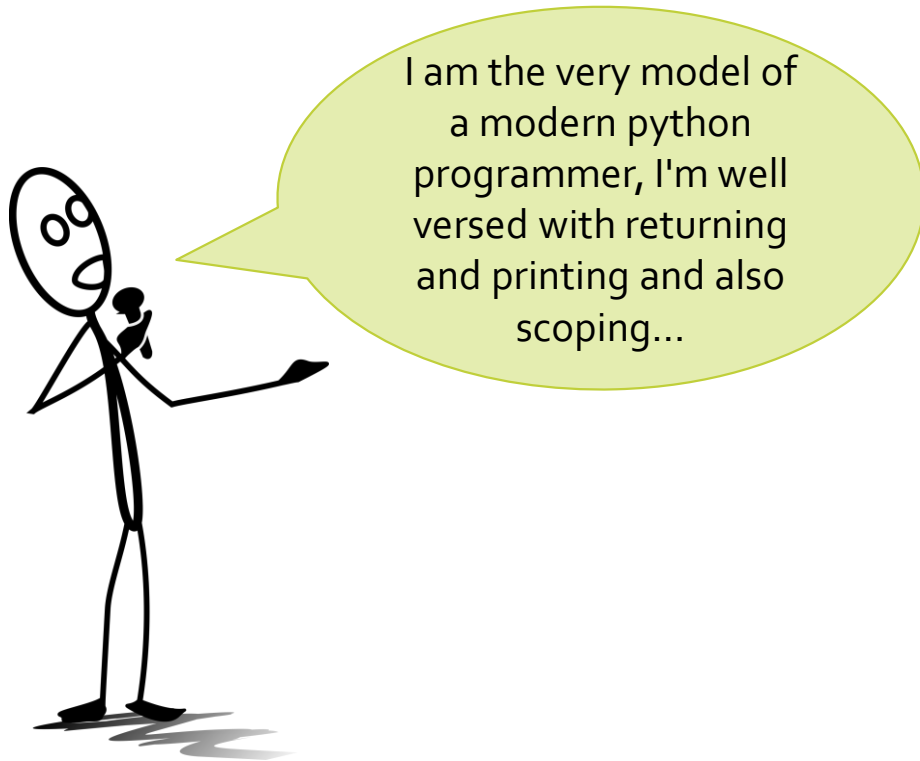
vs.

- Returning: Substitute calling expression with a value



Metaphor

Print



Return



Printing

```
def calculate_grade(grade:int, weight:float)->float:  
    curved = 100 * grade ** .5  
    print(curved)  
    final = curved * weight  
    return final
```

Console

44.0

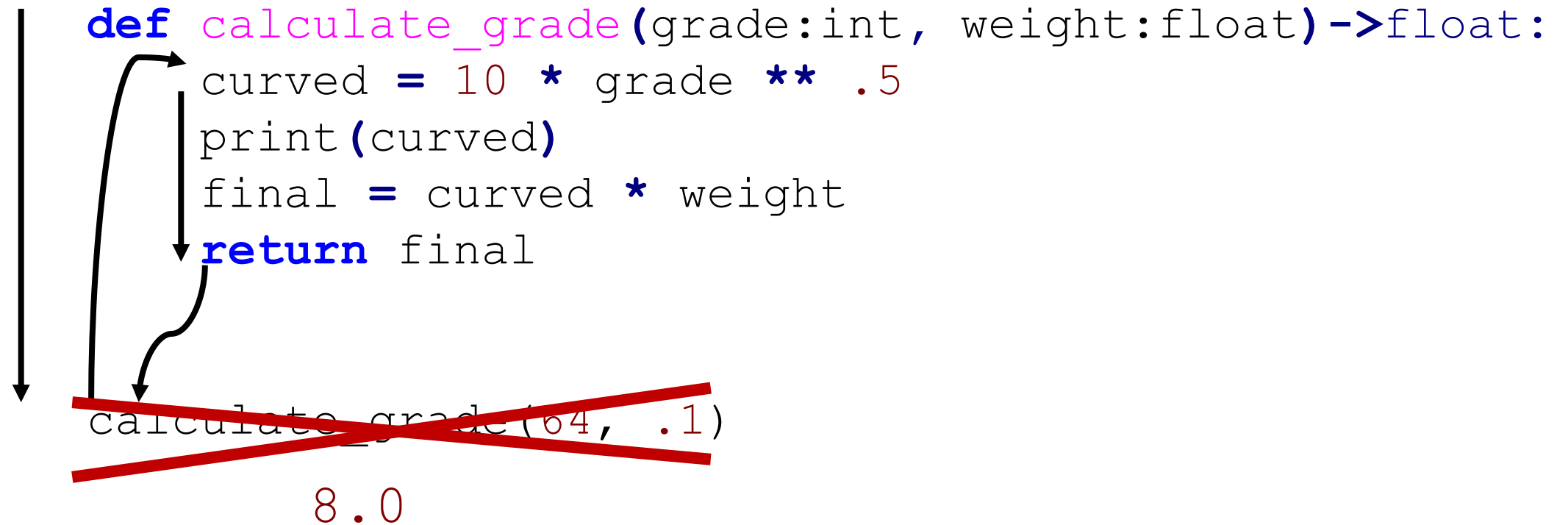


Returning

```
def calculate_grade(grade:int, weight:float)->float:
    curved = 10 * grade ** .5
    print(curved)
    final = curved * weight
    return final
```

~~calculate_grade(64, .1)~~

8.0



The diagram illustrates the execution of the `calculate_grade` function. A vertical arrow on the left points down to the function call `calculate_grade(64, .1)`, which is crossed out with a red X. A curved arrow points from the function call up to the `def` line. Another curved arrow points from the `return final` line down to the function call. Below the function call, the value `8.0` is printed, representing the return value of the function.



Printing without Explicit Return

```
def calculate_grade(grade:int, weight:float)->float:  
    curved = 10 * grade **.5  
    final = curved * weight  
    print(final)
```

```
final_grade = calculate_grade(64, .1)
```

final_grade will be **None**



Syntax

Printing

```
print("Hello World")
```

Needs parentheses

Returning


```
return "A value"
```

No parentheses



When to Return

Write a function that consumes a string and **produces** that string backwards.



Return!



When to Print

Write a function that consumes a string and prints it out backwards

Print!

Write a function that consumes a string and produces that string backwards; then print the result of calling the function

Call, Return, Print

