

Analyzing function calls

Stack diagrams - allow us to visualize the memory and state of the program

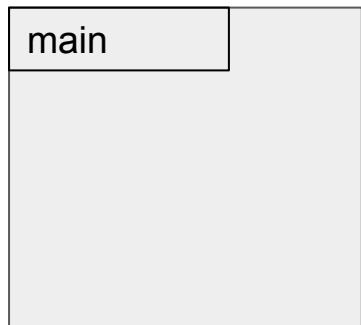
Steps for writing stack diagrams:

1. Pause when you see the function call
2. Create a stack frame for the called function
 - a. parameters allocated
 - b. local variables in scope stored here
3. Copy argument values to parameters
4. Execute the function, line by line, until the return
5. Send back the return value
6. Pop the function off the stack
7. Resume the calling function

Tip: Use Python Tutor to check your answers!

Exercise - add.py

Function Stack



Heap

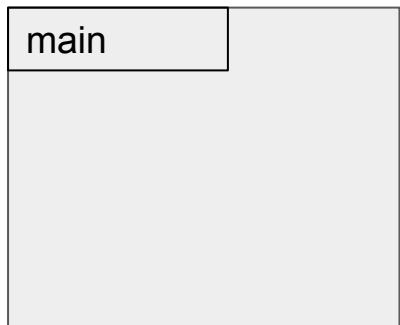
A screenshot of the Atom code editor showing a file named "add.py". The code defines a function "add" that prints its arguments and returns their sum. It also defines a "main" function that calls "add" with arguments 2 and 4, and prints the result. The line "main()" is highlighted with a red rectangle.

```
add.py — ~/CS21/cs21-devel/examples/inclass/w05 — Atom
File Edit View Selection Find Packages Help
add.py add2.py add3.py tip.py fancify.py
1 """
2 A program that adds two numbers using a function
3 """
4
5 def add(x,y):
6     print(x, y)
7     return x + y
8
9 def main():
10    n = 2
11    out = add(n, 4)
12    print(out)
13
14 main()
15
```

add.py 15:1 LF N UTF-8 Python 0 files

Exercise - add.py

Function Stack



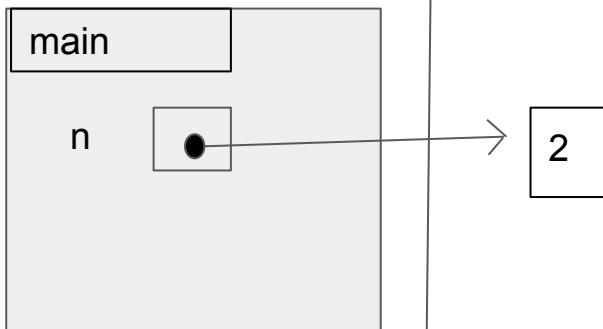
Heap

A screenshot of the Atom code editor showing the file `add.py`. The code defines a function `add` that prints its arguments and returns their sum. It also defines a `main` function that calls `add` with arguments `n` and `4`, and prints the result. A red rectangle highlights the `def main():` line. The status bar at the bottom shows the file is 15:1 long.

```
add.py — ~/CS21/cs21-devel/examples/inclass/w05 — Atom
File Edit View Selection Find Packages Help
add.py add2.py add3.py tip.py fancify.py
1 """
2 A program that adds two numbers using a function
3 """
4
5 def add(x,y):
6     print(x, y)
7     return x + y
8
9 def main():
10     n = 2
11     out = add(n, 4)
12     print(out)
13
14 main()
15
```

Exercise - add.py

Function Stack

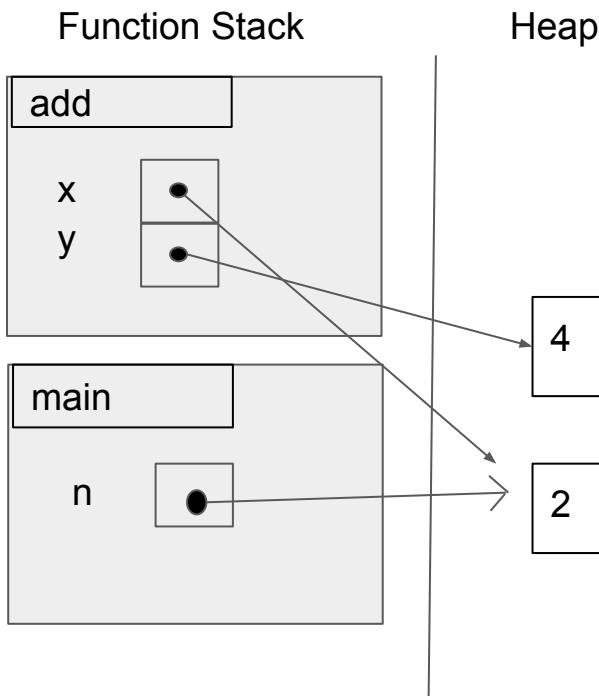


Heap

A screenshot of the Atom code editor showing the file "add.py". The code defines a function "add" that prints its arguments and returns their sum. It also defines a "main" function that calls "add" with arguments 2 and 4, and prints the result. The line "out = add(n, 4)" is highlighted with a red rectangle. The status bar at the bottom shows "add.py 15:1" and "Python".

```
add.py — ~/CS21/cs21-devel/examples/inclass/w05 — Atom
File Edit View Selection Find Packages Help
add.py add2.py add3.py tip.py fancify.py
1 """
2 A program that adds two numbers using a function
3 """
4
5 def add(x,y):
6     print(x, y)
7     return x + y
8
9 def main():
10     n = 2
11     out = add(n, 4)
12     print(out)
13
14 main()
15
```

Exercise - add.py



add.py — ~/CS21/cs21-devel/examples/inclass/w05 — Atom

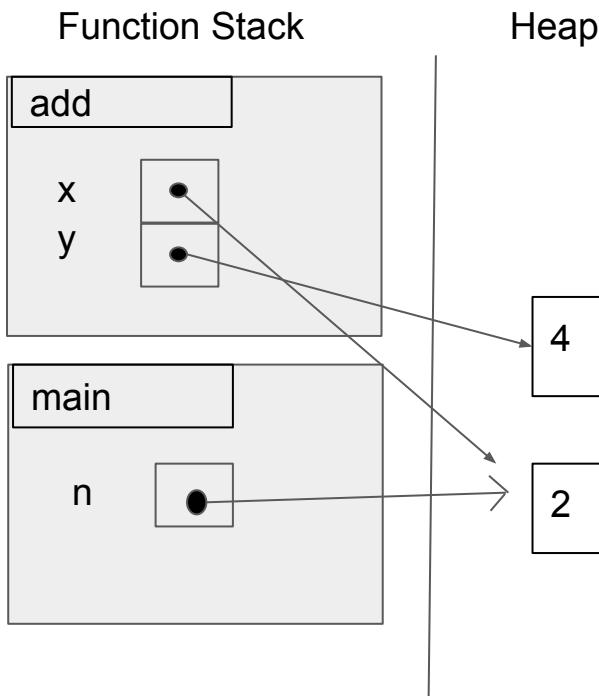
File Edit View Selection Find Packages Help

add.py add2.py add3.py tip.py fancify.py

```
1 """
2 A program that adds two numbers using a function
3 """
4
5 def add(x,y):
6     print(x, y)
7     return x + y
8
9 def main():
10     n = 2
11     out = add(n, 4)
12     print(out)
13
14 main()
15
```

add.py 15:1 LF N UTF-8 Python 0 files

Exercise - add.py



add.py — ~/CS21/cs21-devel/examples/inclass/w05 — Atom

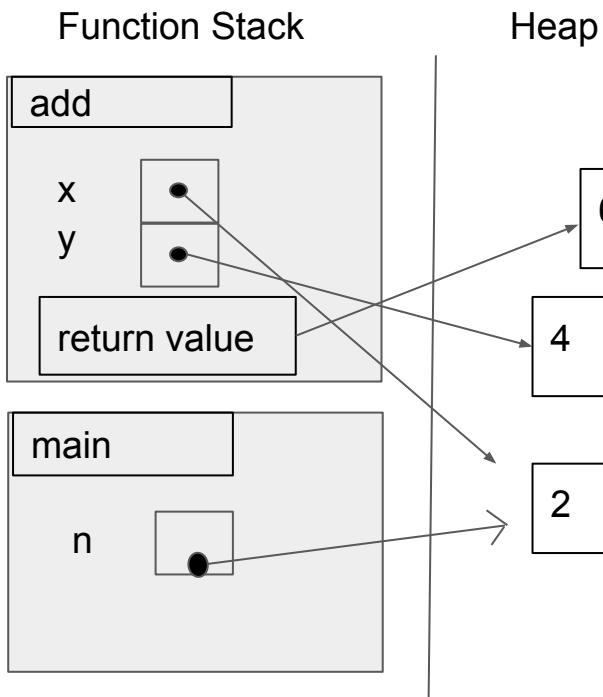
File Edit View Selection Find Packages Help

add.py add2.py add3.py tip.py fancify.py

```
1 """
2 A program that adds two numbers using a function
3 """
4
5 def add(x, y):
6     print(x, y)
7     return x + y
8
9 def main():
10    n = 2
11    out = add(n, 4)
12    print(out)
13
14 main()
15
```

add.py 15:1 LF N UTF-8 Python 0 files

Exercise - add.py



add.py — ~/CS21/cs21-devel/examples/inclass/w05 — Atom

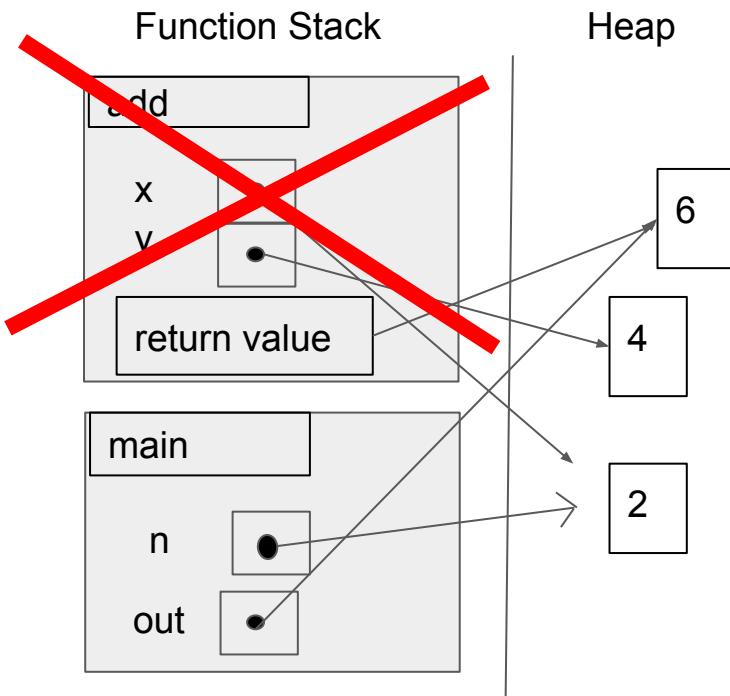
File Edit View Selection Find Packages Help

add.py add2.py add3.py tip.py fancify.py

```
1 """
2 A program that adds two numbers using a function
3 """
4
5 def add(x,y):
6     print(x, y)
7     return x + y
8
9 def main():
10    n = 2
11    out = add(n, 4)
12    print(out)
13
14 main()
15
```

add.py 15:1 LF N UTF-8 Python 0 files

Exercise - add.py



add.py — ~/CS21/cs21-devel/examples/inclass/w05 — Atom

File Edit View Selection Find Packages Help

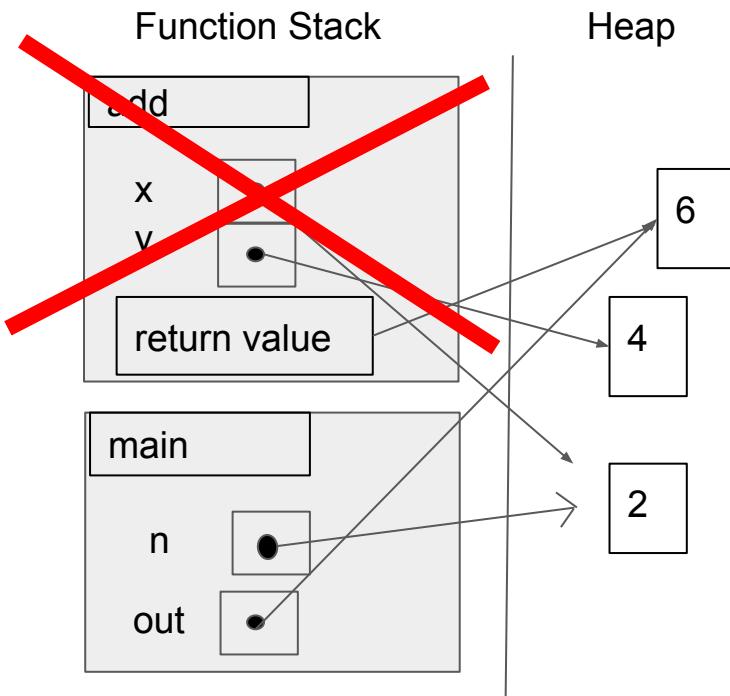
add.py add2.py add3.py tip.py fancify.py

```
1 """
2 A program that adds two numbers using a function
3 """
4
5 def add(x,y):
6     print(x, y)
7     return x + y
8
9 def main():
10     n = 2
11     out = add(n, 4)
12     print(out)
13
14 main()
15
```

add.py 15:1 LF N UTF-8 Python 0 files

A red box highlights the line of code `out = add(n, 4)` in the `main` function, indicating the point of interest for the exercise.

Exercise - add.py



add.py — ~/CS21/cs21-devel/examples/inclass/w05 — Atom

File Edit View Selection Find Packages Help

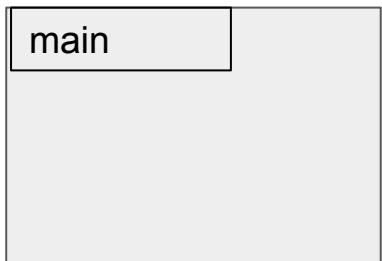
add.py add2.py add3.py tip.py fancify.py

```
1 """
2 A program that adds two numbers using a function
3 """
4
5 def add(x,y):
6     print(x, y)
7     return x + y
8
9 def main():
10    n = 2
11    out = add(n, 4)
12    print(out)
13
14 main()
15
```

add.py 15:1 LF N UTF-8 Python 0 files

Exercise - add2.py

Function Stack



Heap



add2.py — ~/CS21/cs21-devel/examples/inclass/w05 — Atom

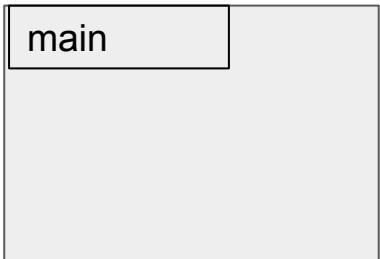
```
File Edit View Selection Find Packages Help
add.py      add2.py      add3.py      tip.py      fancify.py
1 """
2 A program that adds two numbers using a function
3 """
4
5 def add(x,y):
6     print(x, y)
7     return x + y
8
9 def main():
10    x = 2
11    x = add(x, x)
12    print(x)
13
14 main()
15 
```

add2.py 15:1 LF O UTF-8 Python 0 files

Exercise - add2.py

Function Stack

Heap



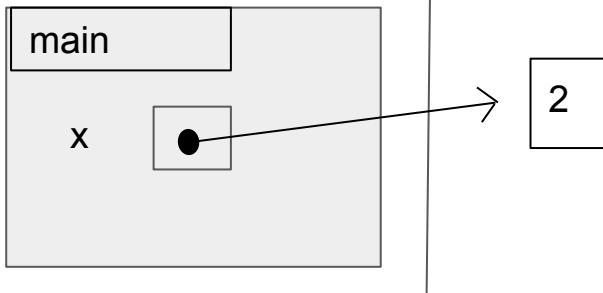
A screenshot of a code editor window titled "add2.py — ~/CS21/cs21-devel/examples/inclass/w05 — Atom". The window shows the following Python code:

```
1 """  
2 A program that adds two numbers using a function  
3 """  
4  
5 def add(x,y):  
6     print(x, y)  
7     return x + y  
8  
9 def main():  
10     x = ?  
11     x = add(x, x)  
12     print(x)  
13  
14 main()  
15 ■
```

The code defines a function `add` that takes two arguments `x` and `y`, prints them, and returns their sum. It also defines a function `main` that initializes `x` to an unknown value, calls `add` with `x` as both arguments, and prints the result. The line `9 def main():` is highlighted with a red box.

Exercise - add2.py

Function Stack



Heap

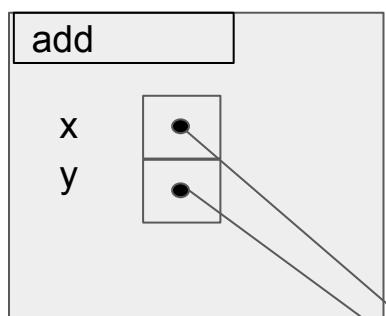
A screenshot of the Atom code editor showing the `add2.py` script. The code defines a function `add` that takes two parameters `x` and `y`, prints them, and returns their sum. It also defines a `main` function that initializes `x` to 2, calls `add` with arguments `x` and `x`, and prints the result. A red rectangle highlights the line `x = 2`.

```
add2.py -- ~/CS21/cs21-devel/examples/inclass/w05 -- Atom
File Edit View Selection Find Packages Help
add.py add2.py add3.py tip.py fancify.py
1 """
2 A program that adds two numbers using a function
3 """
4
5 def add(x,y):
6     print(x, y)
7     return x + y
8
9 def main():
10     x = 2
11     y = add(x, x)
12     print(x)
13
14 main()
15
```

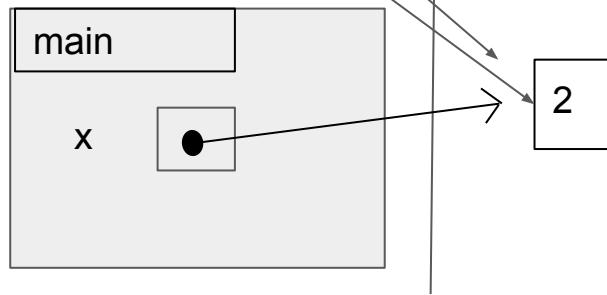
add2.py 15:1 LF O UTF-8 Python 0 files

Exercise - add2.py

Function Stack



Heap



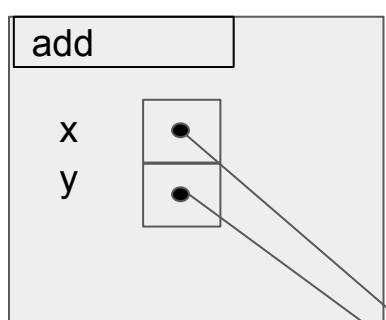
A screenshot of an Atom code editor displaying the file "add2.py". The code is as follows:

```
1 """  
2     A program that adds two numbers using a function  
3 """  
4  
5 def add(x,y):  
6     print(x, y)  
7     return x + y  
8  
9 def main():  
10    x = 2  
11    x = add(x, x)  
12    print(x)  
13  
14 main()  
15 ■
```

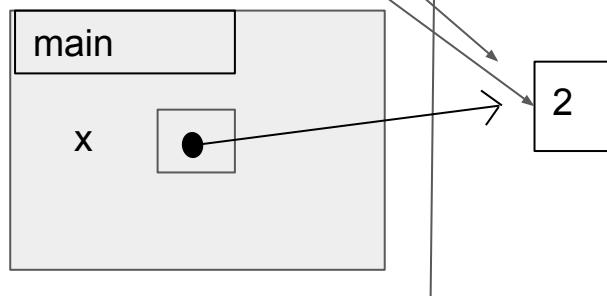
The line "def add(x,y):" is highlighted with a red rectangle. The status bar at the bottom shows "add2.py 15:1" and icons for LF, O, UTF-8, Python, and 0 files.

Exercise - add2.py

Function Stack



Heap



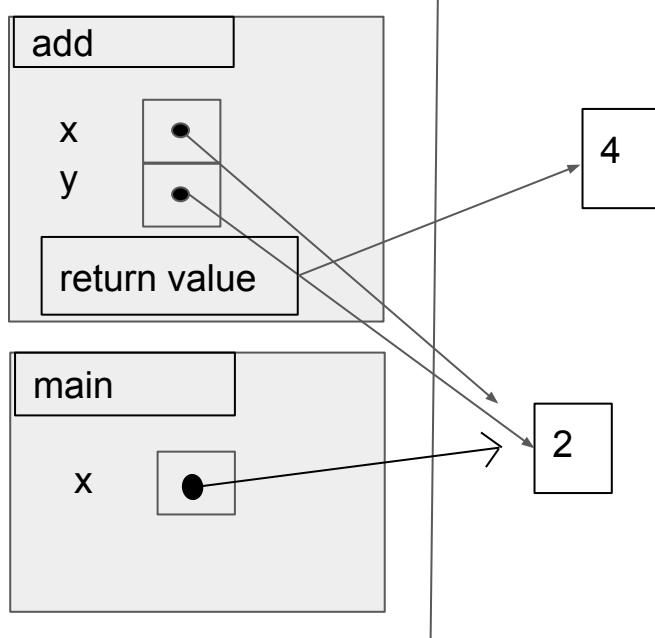
A screenshot of a code editor window titled "add2.py". The window shows the following Python code:

```
1 """  
2 A program that adds two numbers using a function  
3 """  
4  
5 def add(x,y):  
6     print(x, y)  
7     return x + y  
8  
9 def main():  
10    x = 2  
11    x = add(x, x)  
12    print(x)  
13  
14 main()  
15 ■
```

The code defines a function `add` that takes two arguments `x` and `y`, prints them, and returns their sum. It also defines a `main` function that sets `x` to 2, calls `add` with `x` as both arguments, and prints the result. The code editor interface includes tabs for "add.py", "add2.py" (which is the active tab), "add3.py", "tip.py", and "fancify.py". The status bar at the bottom shows "add2.py 15:1" and "LF O UTF-8 Python".

Exercise - add2.py

Function Stack

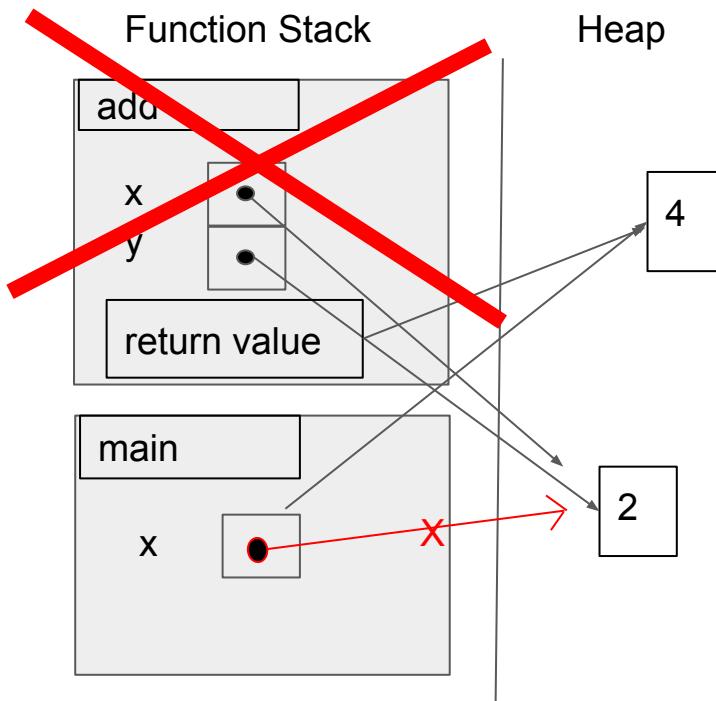


Heap

A screenshot of the Atom code editor displaying the file `add2.py`. The code defines a function `add` that adds its two arguments and returns the result. It then calls this function from the `main` function, passing the value `2` to it twice. The code editor interface includes tabs for other files like `add.py`, `add3.py`, `tip.py`, and `fancify.py`. A red box highlights the body of the `add` function.

```
add2.py — ~/CS21/cs21-devel/examples/inclass/w05 — Atom
File Edit View Selection Find Packages Help
add.py add2.py add3.py tip.py fancify.py
1 """
2 A program that adds two numbers using a function
3 """
4
5 def add(x,y):
6     print(x, y)
7     return x + y
8
9 def main():
10     x = 2
11     x = add(x, x)
12     print(x)
13
14 main()
15
```

Exercise - add2.py

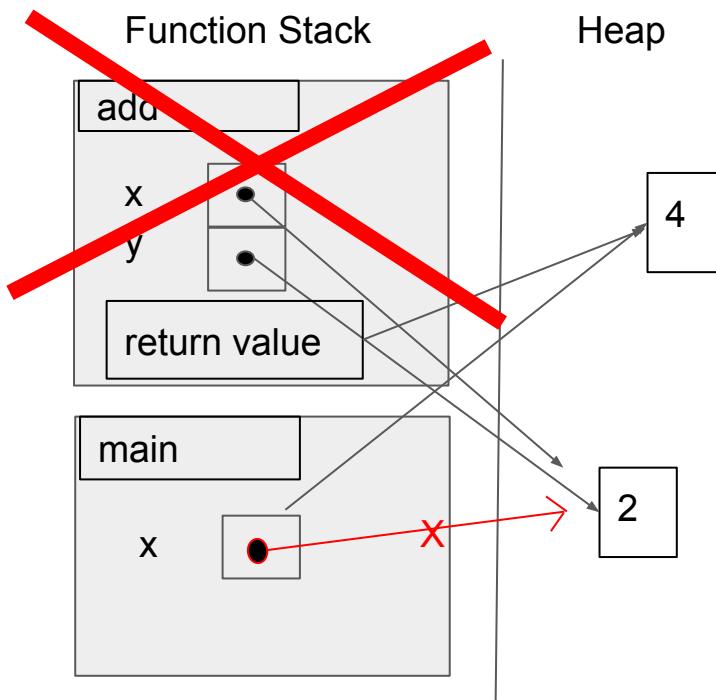


```
add2.py -- ~/CS21/cs21-devel/examples/inclass/w05 -- Atom
File Edit View Selection Find Packages Help
add.py      add2.py      add3.py      tip.py      fancify.py
1 """
2 A program that adds two numbers using a function
3 """
4
5 def add(x,y):
6     print(x, y)
7     return x + y
8
9 def main():
10     x = 2
11     x = add(x, x)
12     print(x)
13
14 main()
15 
```

add2.py 15:1 LF O UTF-8 Python 0 files

A screenshot of a code editor showing the `add2.py` file. The code defines a `main` function that calls the `add` function twice with the same argument `x`. The `add` function prints its arguments and returns their sum. The output of the program is shown in the terminal below the code editor, where it prints `2` and then `4`.

Exercise - add2.py

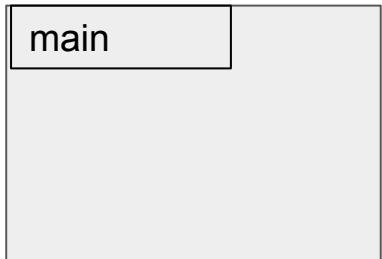


```
add2.py -- ~/CS21/cs21-devel/examples/inclass/w05 -- Atom
File Edit View Selection Find Packages Help
add.py add2.py add3.py tip.py fancify.py
1 """
2 A program that adds two numbers using a function
3 """
4
5 def add(x,y):
6     print(x, y)
7     return x + y
8
9 def main():
10     x = 2
11     x = add(x, x)
12     print(x)
13
14 main()
15 
```

The code in the `add2.py` file defines a function `add` that takes two arguments `x` and `y`, prints them, and returns their sum. It also defines a `main` function that initializes `x` to 2, calls `add(x, x)` to set `x` to 4, and then prints the value of `x`. The code is highlighted with a red rectangle around the `main` function definition.

Exercise - add3.py

Function Stack



Heap

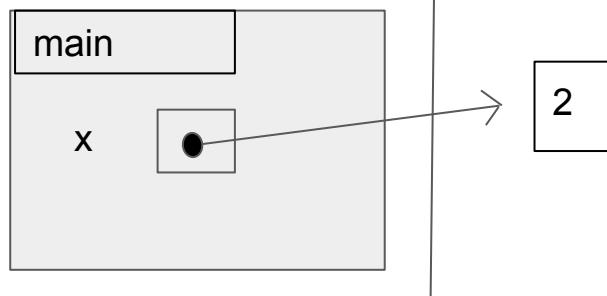
The screenshot shows a code editor window for the file "add3.py". The window title is "add3.py — ~/CS21/cs21-devel/examples/inclass/w05 — Atom". The code is as follows:

```
1 """
2 A program that adds two numbers using a function
3 """
4
5 def add(x,y):
6     x = 10
7     print(x, y)
8     return x + y
9
10 def main():
11     x = 2
12     out = add(x, x)
13     print(x, out)
14
15 main()
16
```

The code editor has tabs for "add.py", "add2.py", "add3.py", "tip.py", "fancify.py", and "addVal.py". The "add3.py" tab is selected. The "main()" function is highlighted with a red rectangle.

Exercise - add3.py

Function Stack



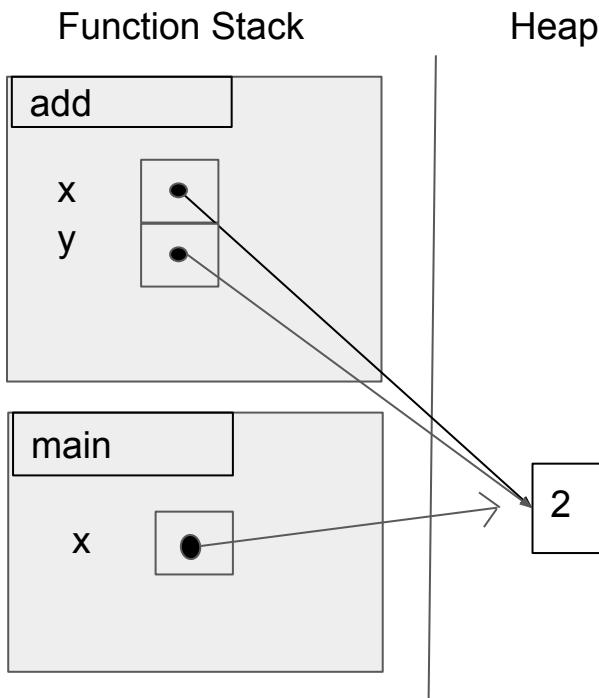
Heap

A screenshot of an Atom code editor window titled `add3.py — ~/CS21/cs21-devel/examples/inclass/w05 — Atom`. The window shows several tabs at the top: `add.py`, `add2.py`, `add3.py` (which is the active tab), `tip.py`, `fancify.py`, and `addVal.py`. The code in the editor is:

```
1 """
2 A program that adds two numbers using a function
3 """
4
5 def add(x,y):
6     x = 10
7     print(x, y)
8     return x + y
9
10 def main():
11     x = 2
12     out = add(x, x)
13     print(x, out)
14
15 main()
16
```

The lines `10` through `13` are highlighted with a red rectangle, indicating the current scope of the variable `x`.

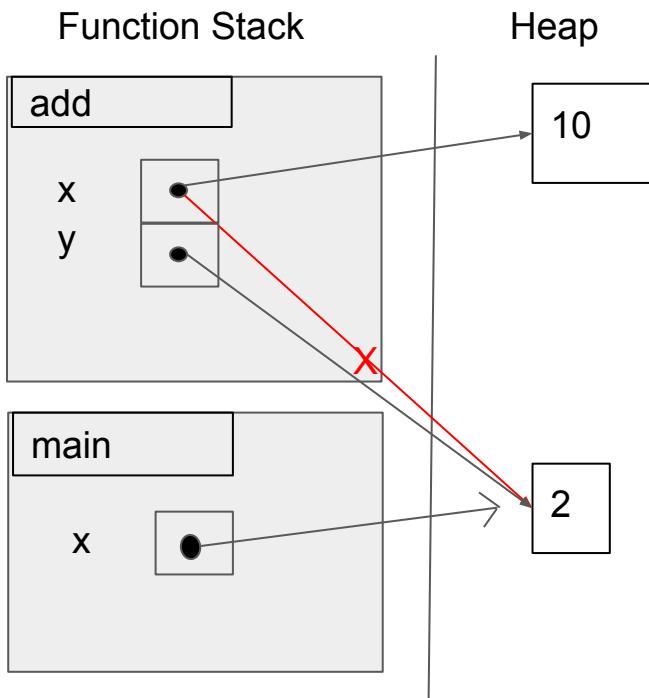
Exercise - add3.py



```
add3.py — ~/CS21/cs21-devel/examples/inclass/w05 — Atom
File Edit View Selection Find Packages Help
add.py add2.py add3.py tip.py fancify.py addVal.py
1 """
2 A program that adds two numbers using a function
3 """
4
5 def add(x,y):
6     x = 10
7     print(x, y)
8     return x + y
9
10 def main():
11     x = 2
12     out = add(x, x)
13     print(x, out)
14
15 main()
16
```

add3.py 8:9 LF N UTF-8 Python 0 files

Exercise - add3.py



add3.py — ~/CS21/cs21-devel/examples/inclass/w05 — Atom

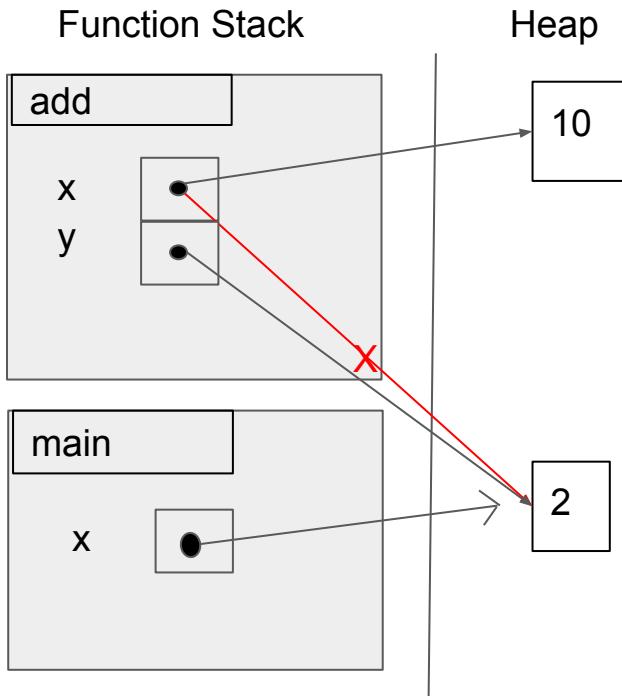
File Edit View Selection Find Packages Help

add.py add2.py add3.py tip.py fancify.py addVal.py

```
1 """
2 A program that adds two numbers using a function
3 """
4
5 def add(x,y):
6     x = 10
7     print(x, y)
8     return x + y
9
10 def main():
11     x = 2
12     out = add(x, x)
13     print(x, out)
14
15 main()
16
```

add3.py 8:9 LF N UTF-8 Python 0 files

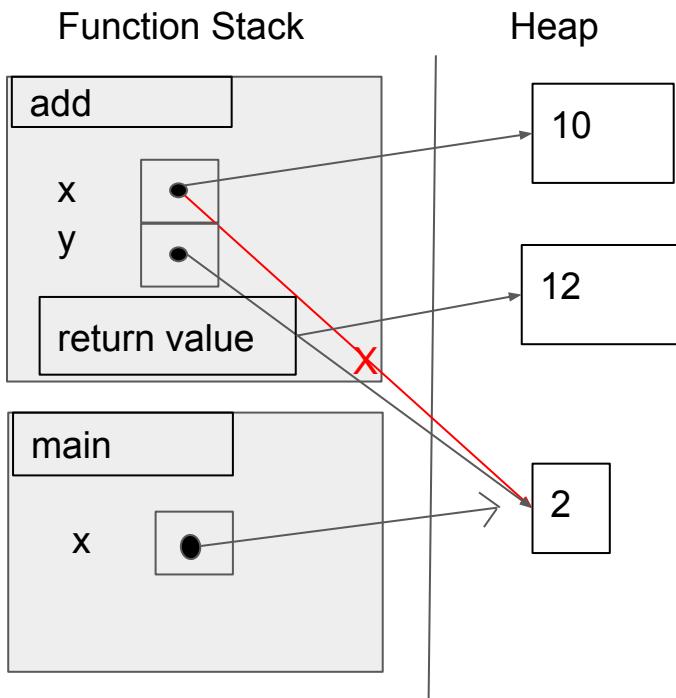
Exercise - add3.py



```
add3.py — ~/CS21/cs21-devel/examples/inclass/w05 — Atom
File Edit View Selection Find Packages Help
add.py add2.py add3.py tip.py fancify.py addVal.py
1 """
2 A program that adds two numbers using a function
3 """
4
5 def add(x,y):
6     x = 10
7     print(x, y)
8     return x + y
9
10 def main():
11     x = 2
12     out = add(x, x)
13     print(x, out)
14
15 main()
16
```

add3.py 8:9 LF N UTF-8 Python 0 files

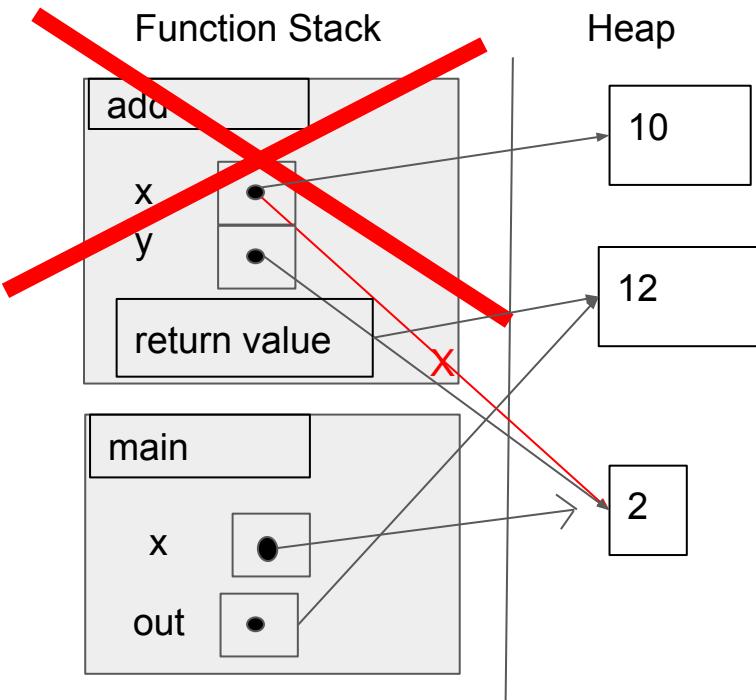
Exercise - add3.py



```
add3.py — ~/CS21/cs21-devel/examples/inclass/w05 — Atom
File Edit View Selection Find Packages Help
add.py add2.py add3.py tip.py fancify.py addVal.py
1 """
2 A program that adds two numbers using a function
3 """
4
5 def add(x,y):
6     x = 10
7     print(x + y)
8     return x + y
9
10 def main():
11     x = 2
12     out = add(x, x)
13     print(x, out)
14
15 main()
16
```

add3.py 8:9 LF N UTF-8 Python 0 files

Exercise - add3.py



add3.py — ~/CS21/cs21-devel/examples/inclass/w05 — Atom

File Edit View Selection Find Packages Help

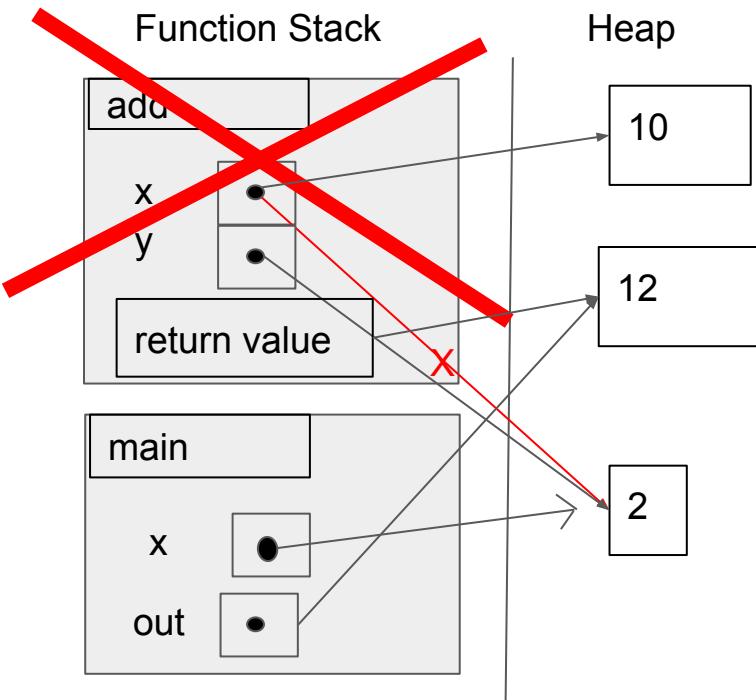
add.py add2.py add3.py tip.py fancify.py addVal.py

```
1 """
2 A program that adds two numbers using a function
3 """
4
5 def add(x,y):
6     x = 10
7     print(x, y)
8     return x + y
9
10 def main():
11     x = 2
12     out = add(x, x)
13     print(x, out)
14
15 main()
16
```

Note that x has the same value in main() after add() returns!

add3.py 8:9 LF N UTF-8 Python 0 files

Exercise - add3.py



add3.py — ~/CS21/cs21-devel/examples/inclass/w05 — Atom

File Edit View Selection Find Packages Help

add.py add2.py add3.py tip.py fancify.py addVal.py

```
1 """
2 A program that adds two numbers using a function
3 """
4
5 def add(x,y):
6     x = 10
7     print(x, y)
8     return x + y
9
10 def main():
11     x = 2
12     out = add(x, x)
13     print(x, out)
14
15 main()
16
```

Note that x has the same value in main() after add() returns!

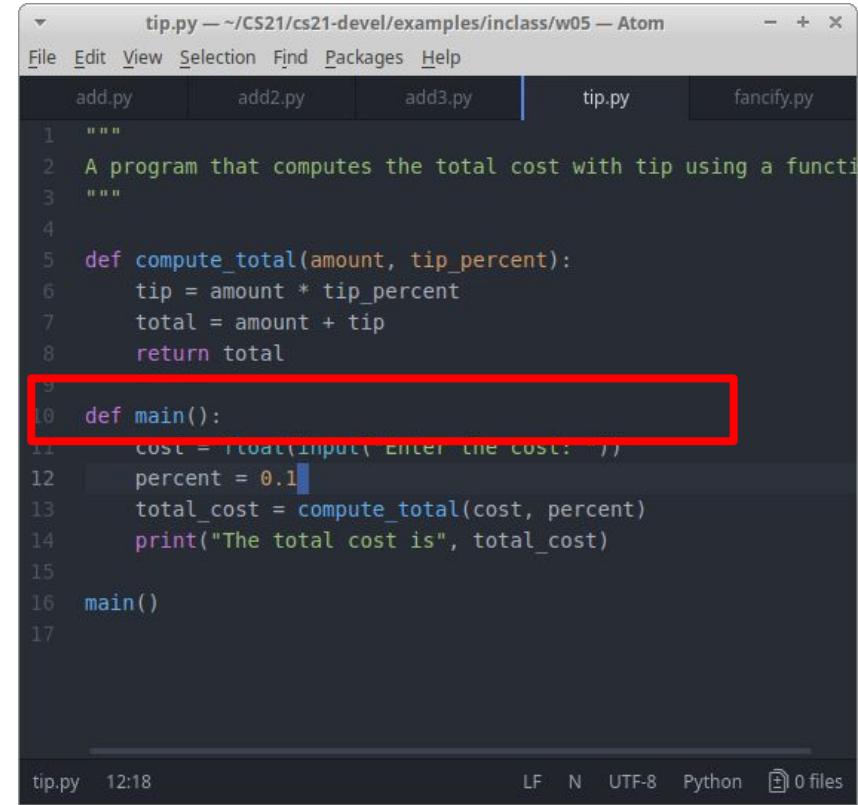
add3.py 8:9 LF N UTF-8 Python 0 files

Exercise - tip.py

Function Stack

Heap

main

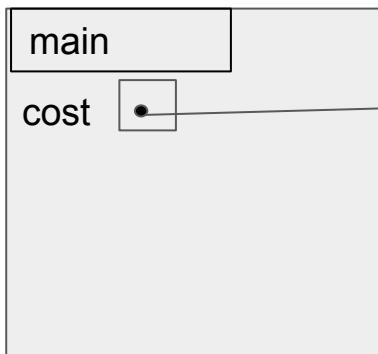


```
tip.py — ~/CS21/cs21-devel/examples/inclass/w05 — Atom
File Edit View Selection Find Packages Help
add.py add2.py add3.py tip.py fancify.py
1 """
2 A program that computes the total cost with tip using a function
3 """
4
5 def compute_total(amount, tip_percent):
6     tip = amount * tip_percent
7     total = amount + tip
8     return total
9
10 def main():
11     cost = float(input("Enter the cost: "))
12     percent = 0.1
13     total_cost = compute_total(cost, percent)
14     print("The total cost is", total_cost)
15
16 main()
17
```

tip.py 12:18 LF N UTF-8 Python 0 files

Exercise - tip.py

Function Stack



Heap

tip.py — ~/CS21/cs21-devel/examples/inclass/w05 — Atom

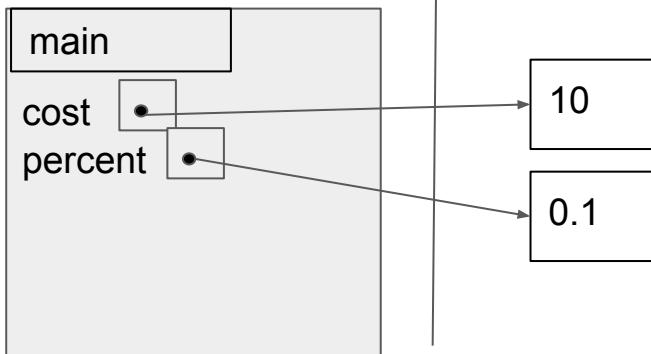
```
File Edit View Selection Find Packages Help
add.py add2.py add3.py tip.py fancify.py
1 """
2 A program that computes the total cost with tip using a function
3 """
4
5 def compute_total(amount, tip_percent):
6     tip = amount * tip_percent
7     total = amount + tip
8     return total
9
10 def main():
11     cost = float(input("Enter the cost: "))
12     percent = 0.1
13     total_cost = compute_total(cost, percent)
14     print("The total cost is", total_cost)
15
16 main()
17
```

tip.py 12:18 LF N UTF-8 Python 0 files

Assume
cost = 10

Exercise - tip.py

Function Stack

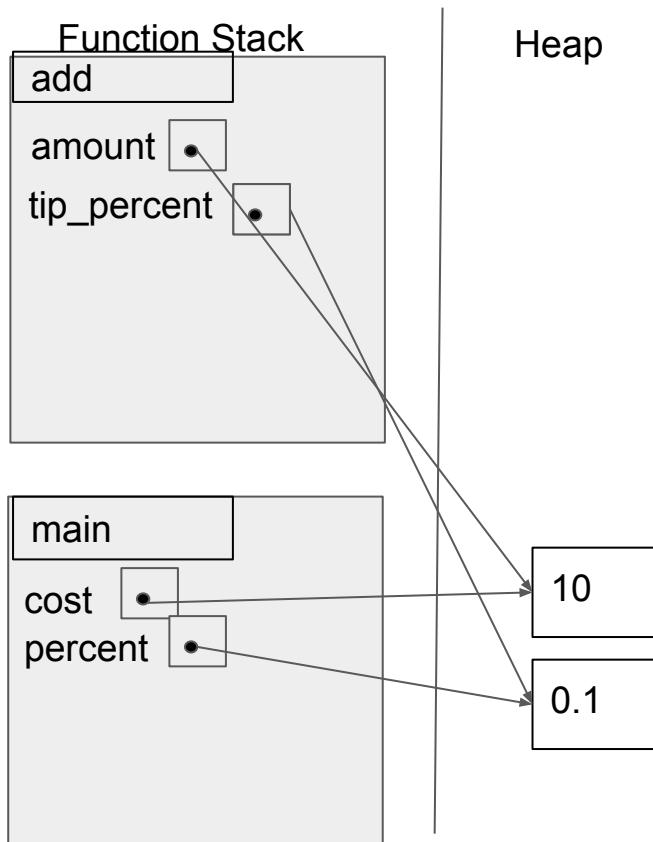


Heap

A screenshot of the Atom code editor showing the `tip.py` script. The code defines a `compute_total` function that takes `amount` and `tip_percent` as arguments, calculates the tip, and returns the total cost. It also defines a `main` function that prompts the user for the cost, sets the tip percent to 0.1, calls `compute_total`, and prints the result. The line `total_cost = compute_total(cost, percent)` is highlighted with a red rectangle.

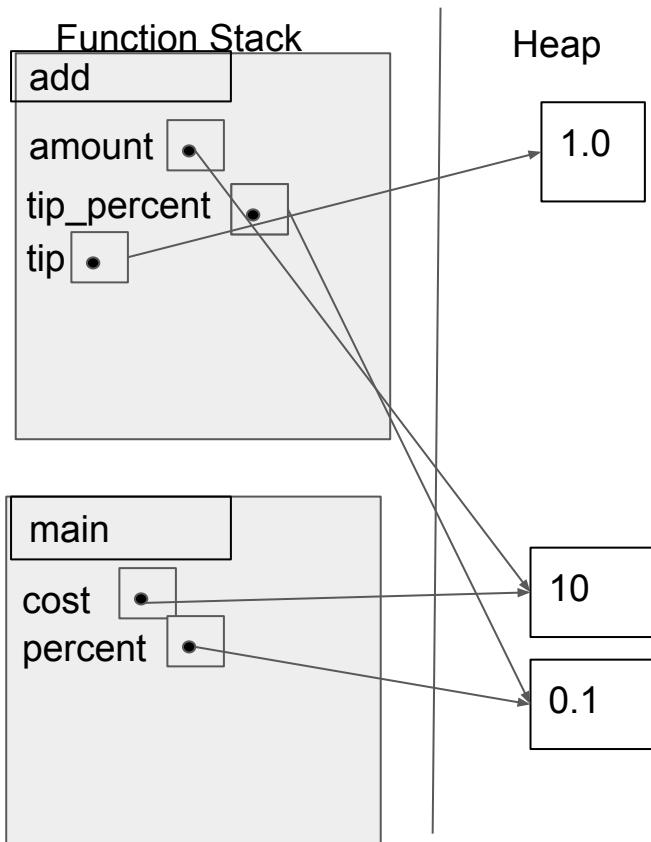
```
tip.py -- ~/CS21/cs21-devel/examples/inclass/w05 -- Atom
File Edit View Selection Find Packages Help
add.py add2.py add3.py tip.py fancify.py
1 """
2 A program that computes the total cost with tip using a function
3 """
4
5 def compute_total(amount, tip_percent):
6     tip = amount * tip_percent
7     total = amount + tip
8     return total
9
10 def main():
11     cost = float(input("Enter the cost: "))
12     percent = 0.1
13     total_cost = compute_total(cost, percent)
14     print("The total cost is", total_cost)
15
16 main()
17
```

Exercise - tip.py



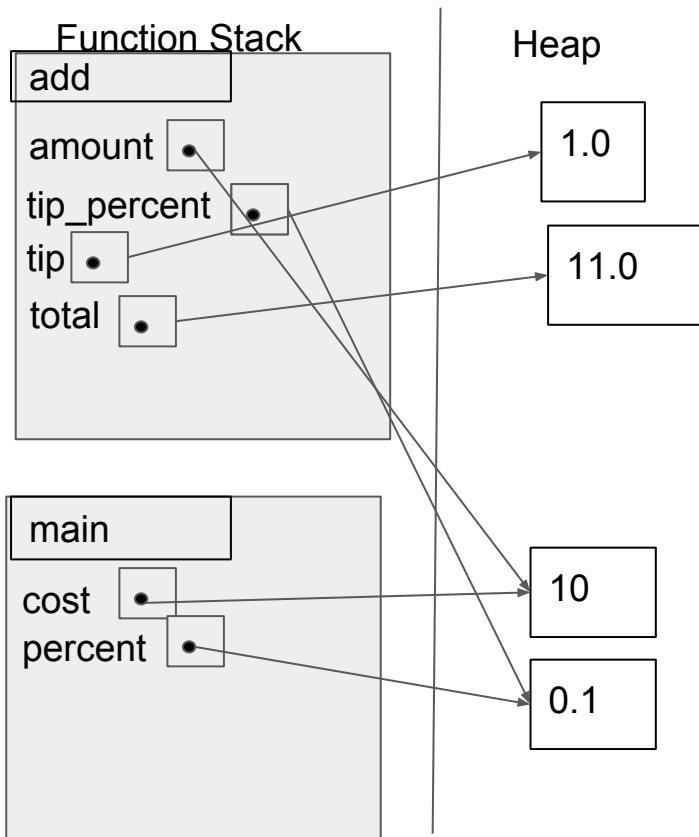
```
tip.py -- ~/CS21/cs21-devel/examples/inclass/w05 -- Atom
File Edit View Selection Find Packages Help
add.py add2.py add3.py tip.py fancify.py
1 """
2 A program that computes the total cost with tip using a function
3 """
4
5 def compute_total(amount, tip_percent):
6     tip = amount * tip_percent
7     total = amount + tip
8     return total
9
10 def main():
11     cost = float(input("Enter the cost: "))
12     percent = 0.1
13     total_cost = compute_total(cost, percent)
14     print("The total cost is", total_cost)
15
16 main()
17
```

Exercise - tip.py



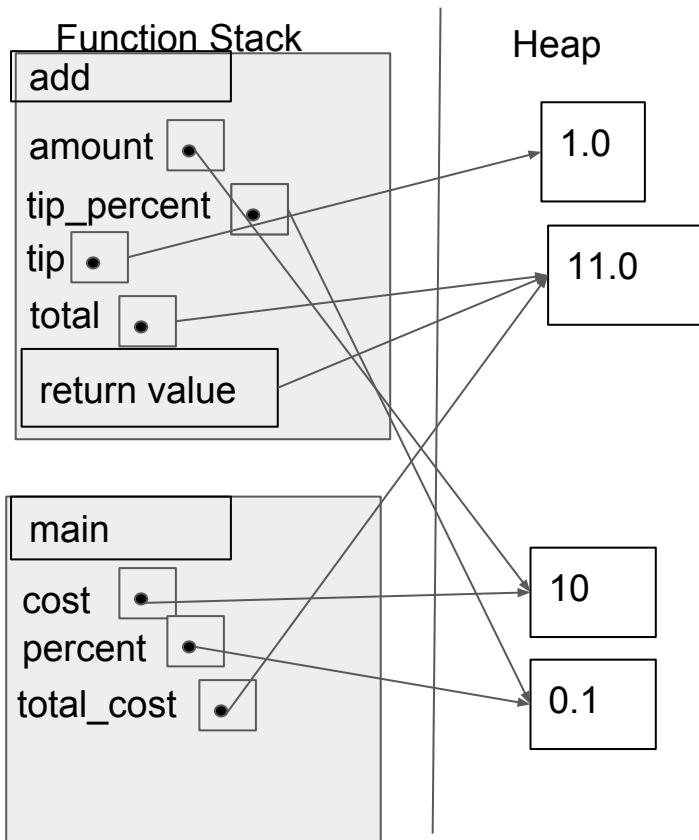
```
tip.py -- ~/CS21/cs21-devel/examples/inclass/w05 -- Atom
File Edit View Selection Find Packages Help
add.py add2.py add3.py tip.py fancify.py
1 """
2 A program that computes the total cost with tip using a function
3 """
4
5 def compute_total(amount, tip_percent):
6     tip = amount * tip_percent
7     total = amount + tip
8     return total
9
10 def main():
11     cost = float(input("Enter the cost: "))
12     percent = 0.1
13     total_cost = compute_total(cost, percent)
14     print("The total cost is", total_cost)
15
16 main()
17
```

Exercise - tip.py



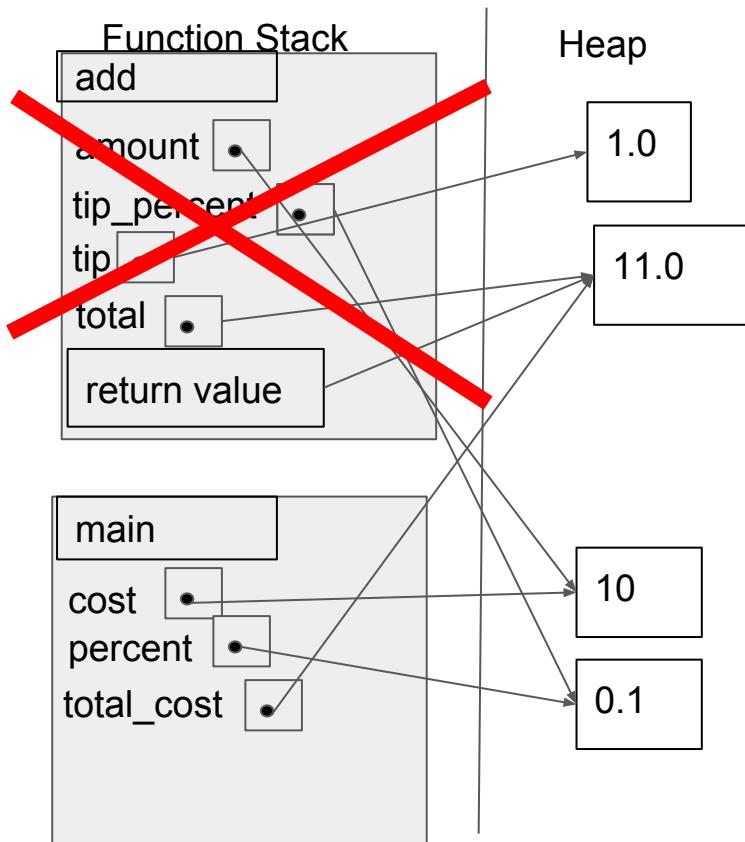
```
tip.py -- ~/CS21/cs21-devel/examples/inclass/w05 -- Atom
File Edit View Selection Find Packages Help
add.py add2.py add3.py tip.py fancify.py
1 """
2 A program that computes the total cost with tip using a function
3 """
4
5 def compute_total(amount, tip_percent):
6     tip = amount * tip_percent
7     total = amount + tip
8     return total
9
10 def main():
11     cost = float(input("Enter the cost: "))
12     percent = 0.1
13     total_cost = compute_total(cost, percent)
14     print("The total cost is", total_cost)
15
16 main()
17
```

Exercise - tip.py



```
tip.py -- ~/CS21/cs21-devel/examples/inclass/w05 -- Atom
File Edit View Selection Find Packages Help
add.py add2.py add3.py tip.py fancify.py
1 """
2 A program that computes the total cost with tip using a function
3 """
4
5 def compute_total(amount, tip_percent):
6     tip = amount * tip_percent
7     total = amount + tip
8     return total
9
10 def main():
11     cost = float(input("Enter the cost: "))
12     percent = 0.1
13     total_cost = compute_total(cost, percent)
14     print("The total cost is", total_cost)
15
16 main()
17
```

Exercise - tip.py



tip.py — ~/CS21/cs21-devel/examples/inclass/w05 — Atom

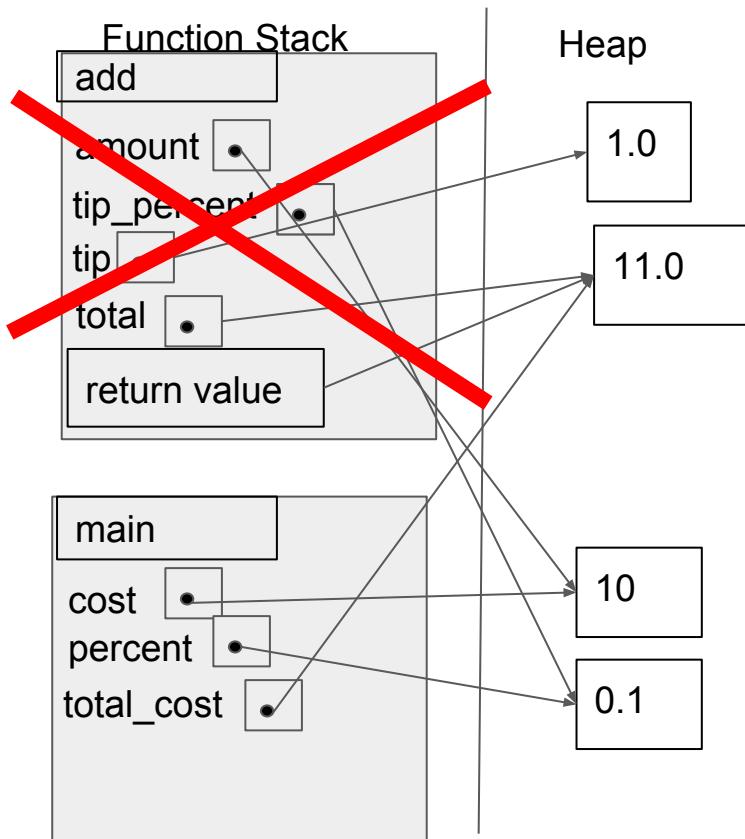
File Edit View Selection Find Packages Help

add.py add2.py add3.py tip.py fancify.py

```
1 """
2 A program that computes the total cost with tip using a function
3 """
4
5 def compute_total(amount, tip_percent):
6     tip = amount * tip_percent
7     total = amount + tip
8     return total
9
10 def main():
11     cost = float(input("Enter the cost: "))
12     percent = 0.1
13     total_cost = compute_total(cost, percent)
14     print("The total cost is", total_cost)
15
16 main()
17
```

tip.py 12:18 LF N UTF-8 Python 0 files

Exercise - tip.py



tip.py — ~/CS21/cs21-devel/examples/inclass/w05 — Atom

```
File Edit View Selection Find Packages Help
add.py add2.py add3.py tip.py fancify.py

1 """
2 A program that computes the total cost with tip using a function
3 """

4
5 def compute_total(amount, tip_percent):
6     tip = amount * tip_percent
7     total = amount + tip
8     return total
9
10 def main():
11     cost = float(input("Enter the cost: "))
12     percent = 0.1
13     total_cost = compute_total(cost, percent)
14     print("The total cost is", total_cost)
15
16 main()
17
```

tip.py 12:18 LF N UTF-8 Python 0 files