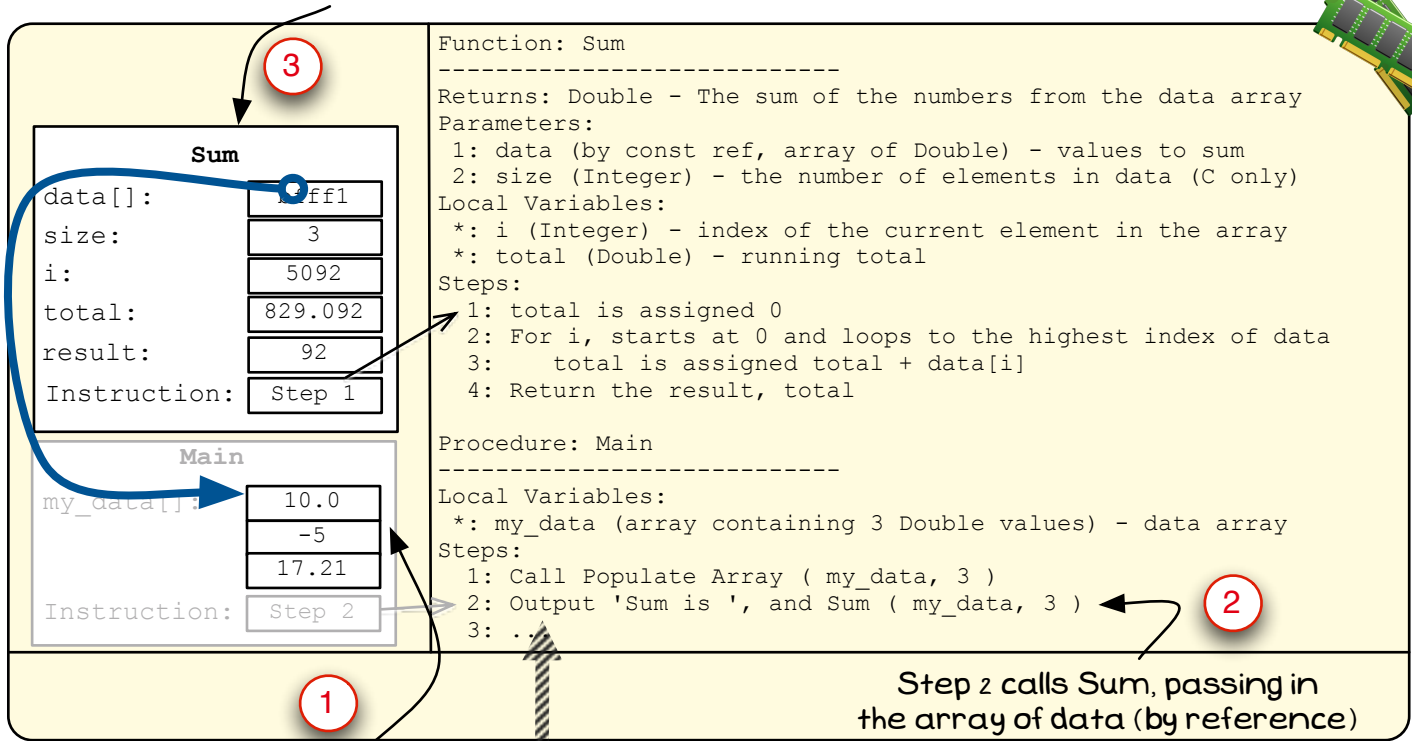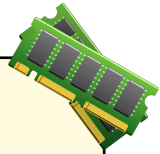Sum is called,
data is passed the address of my_data
(in C size is passed 3)
and space is allocated for i, and total

(3)

Sum

data[]:	kff1
size:	3
i:	5092
total:	829.092
result:	92
Instruction:	Step 1

Main

my_data[]:	10.0
	-5
	17.21
Instruction:	Step 2

```
Function: Sum
----------------------------
Returns: Double - The sum of the numbers from the data array
Parameters:
 1: data (by const ref, array of Double) - values to sum
 2: size (Integer) - the number of elements in data (C only)
Local Variables:
 *: i (Integer) - index of the current element in the array
 *: total (Double) - running total
Steps:
 1: total is assigned 0
 2: For i, starts at 0 and loops to the highest index of data
 3:    total is assigned total + data[i]
 4: Return the result, total

Procedure: Main
----------------------------
Local Variables:
 *: my_data (array containing 3 Double values) - data array
Steps:
 1: Call Populate Array ( my_data, 3 )
 2: Output 'Sum is ', and Sum ( my_data, 3 )
 3: ..
```

(2)

Step 2 calls Sum, passing in
the array of data (by reference)

(1)

Step 1 of Main populated the
array with values from the
user.

Terminal — bash — 80×24

Enter value 1: Ten
Please enter a number.
Enter value 1: 10.0
Enter value 2: -5
Enter value 3: 17.21