Recap

- From lecture 8
  - write functions which perform a task
  - understand that a function can call another function
  - understand the scope of variables inside the function
  - always use excellent function names and variable names to ensure that the purpose of the function is clear

```python
def get_discount(amount, rate):
    discount = amount * rate / 100
    return round(discount, 2)

def get_discount_message(discount, rate):
    rate_message = str(rate) + "%"
    message = rate_message + " Discount: $" + str(discount)
    return message

def print_docket(cost, discount_rate):
    #Code not shown here

print_docket(234, 5)
print_docket(657, 15)
```

**Original price $234**
5% Discount: $11.7
Price $222.3

**Original price $657**
15% Discount: $98.55
Price $558.45
def function1():
    print("A")
    function2(3)
    print("B")

def function2(num):
    print("C")
    function4(num - 1, num - 2)
    print("D")

def function3(number):
    print("E", number)

def function4(num1, num2):
    print("F")
    function3(num1 + num2)

def main():
    print("G")
    function1()
    main()
```python
def get_part(digits, start, end):
    num = int(digits[start: end])
    return num

def number_fiddle(digit_str, number_of_digits):
    part_way = number_of_digits // 2
    part1 = get_part(digit_str, 0, part_way)
    part2 = get_part(digit_str, part_way, number_of_digits)
    return part1 + part2

def display_results(num1, num2):
    print(num1,",",num2, sep = "")

def main():
    num = 3271
    fiddled = number_fiddle(str(num), len(str(num)))
    display_results(num - 5, fiddled)
main()
```
Complete the code trace of the program and show the output.

```python
def first(a):
    b = 3
    print("1", a)
    return second(a * b) + b

def second(a):
    print("2", a)
    return a % 4

def main():
    a = 5
    b = first(a)
    print("3", b)
    b = second(b)
    print("4", b)

main()
```

Output:

```
1 5
2 15
3 6
2 6
4 2
```