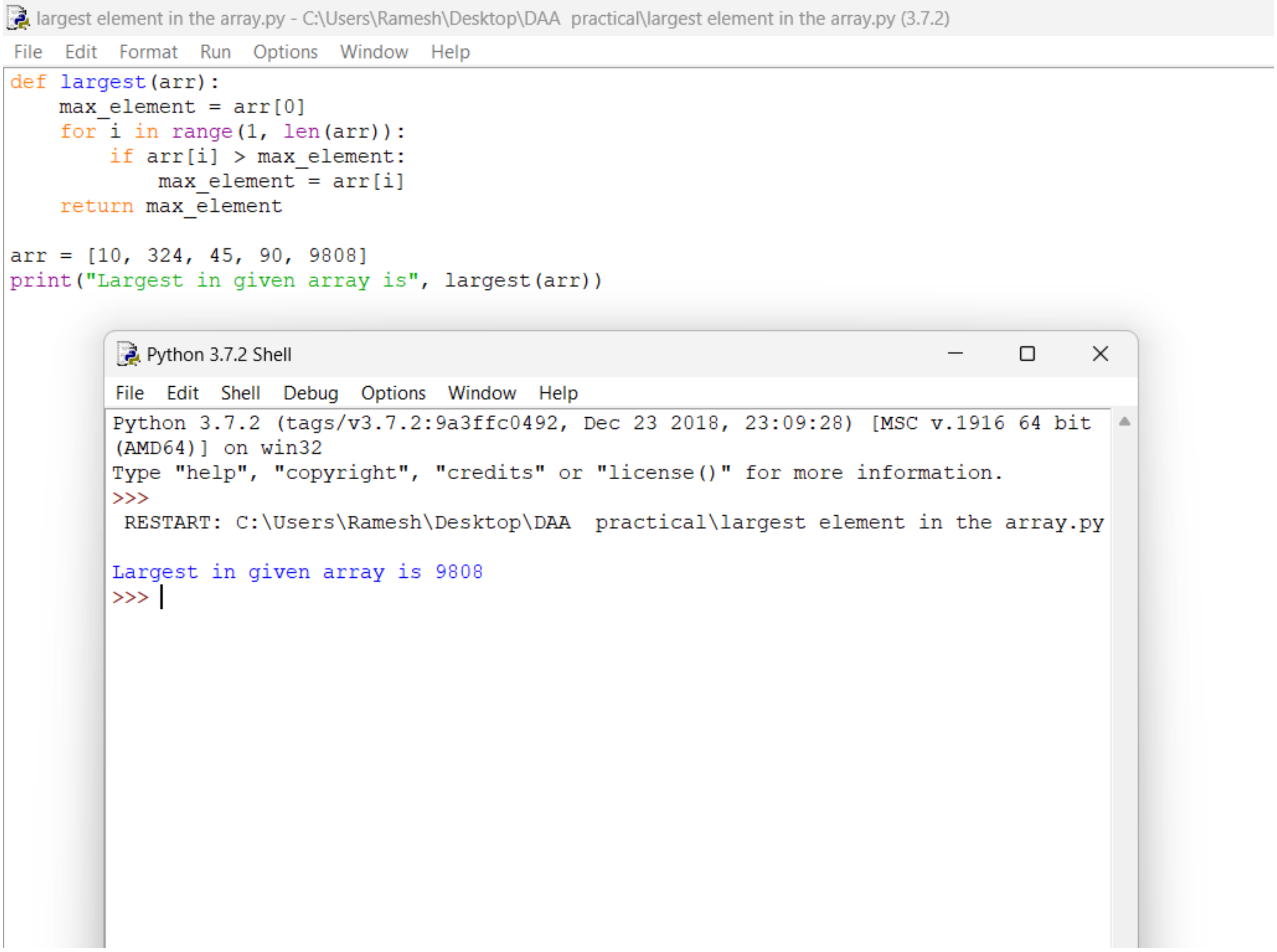
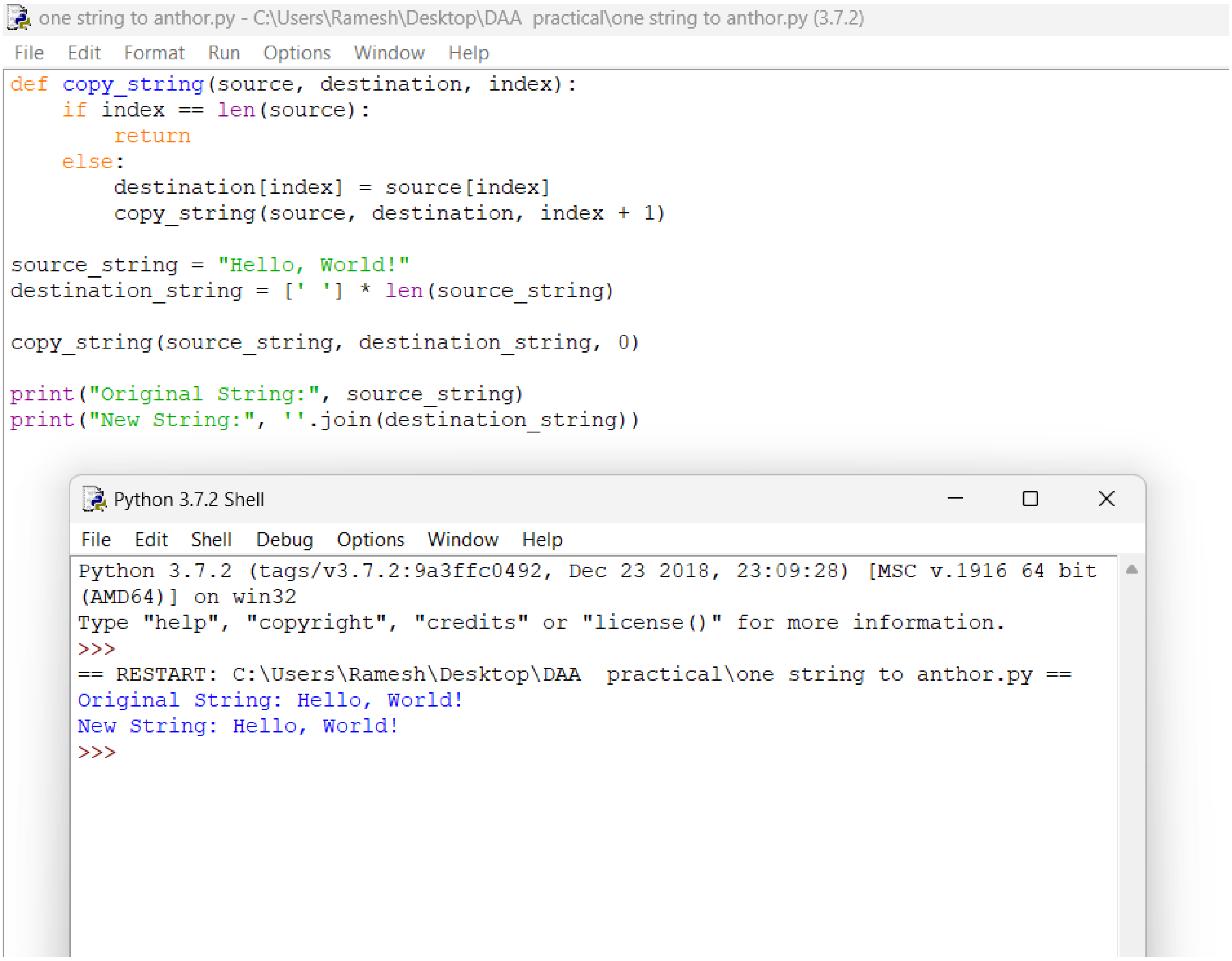

largest element in the array.py - C:\Users\Ramesh\Desktop\DAA practical\largest element in the array.py (3.7.2)
File Edit Format Run Options Window Help
Python 3.7.2 Shell
File Edit Shell Debug Options Window Help

```
def largest(arr):
    max_element = arr[0]
    for i in range(1, len(arr)):
        if arr[i] > max_element:
            max_element = arr[i]
    return max_element

arr = [10, 324, 45, 90, 9808]
print("Largest in given array is", largest(arr))
```

```
Python 3.7.2 (tags/v3.7.2:9a3ffc0492, Dec 23 2018, 23:09:28) [MSC v.1916 64 bit
(AMD64)] on win32
Type "help", "copyright", "credits" or "license()" for more information.
>>> 
 RESTART: C:\Users\Ramesh\Desktop\DAA  practical\largest element in the array.py

Largest in given array is 9808
>>> 
```


one string to anthor.py - C:\Users\Ramesh\Desktop\DAA practical\one string to anthor.py (3.7.2)
File Edit Format Run Options Window Help
Python 3.7.2 Shell
File Edit Shell Debug Options Window Help

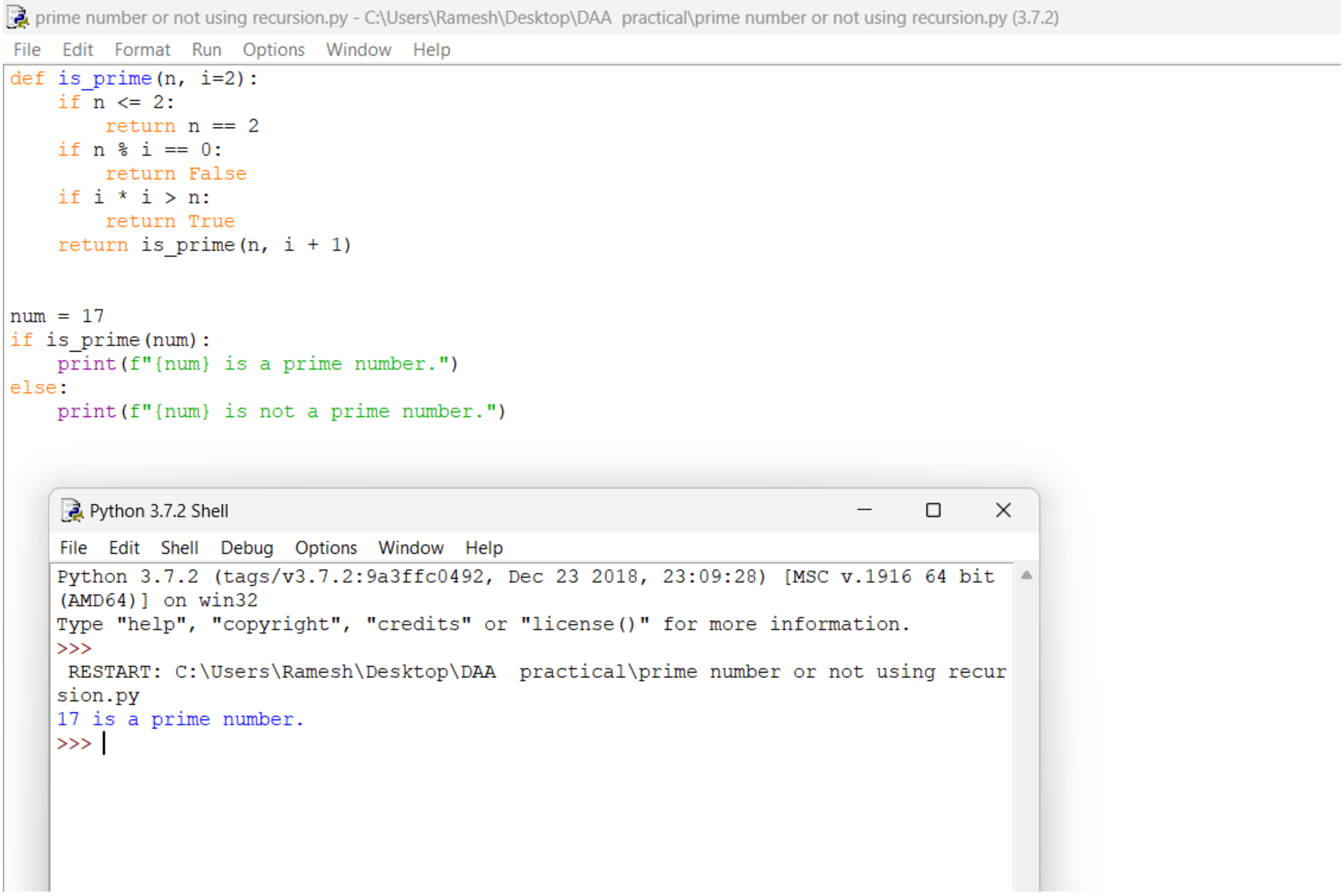
```python
def copy_string(source, destination, index):
    if index == len(source):
        return
    else:
        destination[index] = source[index]
        copy_string(source, destination, index + 1)

source_string = "Hello, World!"
destination_string = [' '] * len(source_string)

copy_string(source_string, destination_string, 0)

print("Original String:", source_string)
print("New String:", ''.join(destination_string))
```

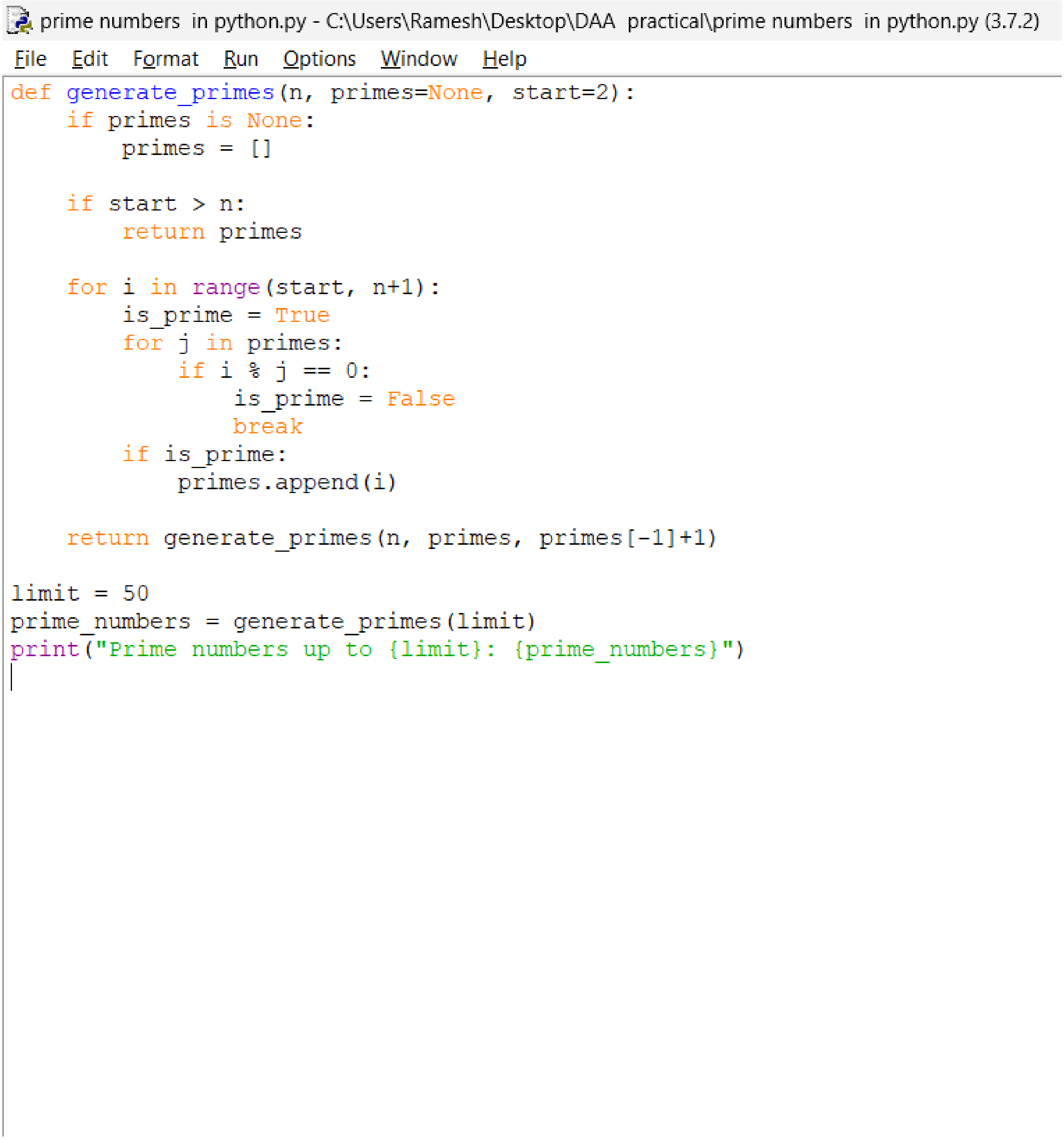
```
Python 3.7.2 (tags/v3.7.2:9a3ffc0492, Dec 23 2018, 23:09:28) [MSC v.1916 64 bit
(AMD64)] on win32
Type "help", "copyright", "credits" or "license()" for more information.
>>>
== RESTART: C:\Users\Ramesh\Desktop\DAA  practical\one string to anthor.py ==
Original String: Hello, World!
New String: Hello, World!
>>>
```


prime number or not using recursion.py - C:\Users\Ramesh\Desktop\DAA practical\prime number or not using recursion.py (3.7.2)
File Edit Format Run Options Window Help
Python 3.7.2 Shell
File Edit Shell Debug Options Window Help

```
def is_prime(n, i=2):
    if n <= 2:
        return n == 2
    if n % i == 0:
        return False
    if i * i > n:
        return True
    return is_prime(n, i + 1)


num = 17
if is_prime(num):
    print(f"{num} is a prime number.")
else:
    print(f"{num} is not a prime number.")
```

```
Python 3.7.2 (tags/v3.7.2:9a3ffc0492, Dec 23 2018, 23:09:28) [MSC v.1916 64 bit
(AMD64)] on win32
Type "help", "copyright", "credits" or "license()" for more information.
>>>
 RESTART: C:\Users\Ramesh\Desktop\DAA  practical\prime number or not using recur
sion.py
17 is a prime number.
>>> 
```



prime numbers in python.py - C:\Users\Ramesh\Desktop\DAA practical\prime numbers in python.py (3.7.2)
File Edit Format Run Options Window Help

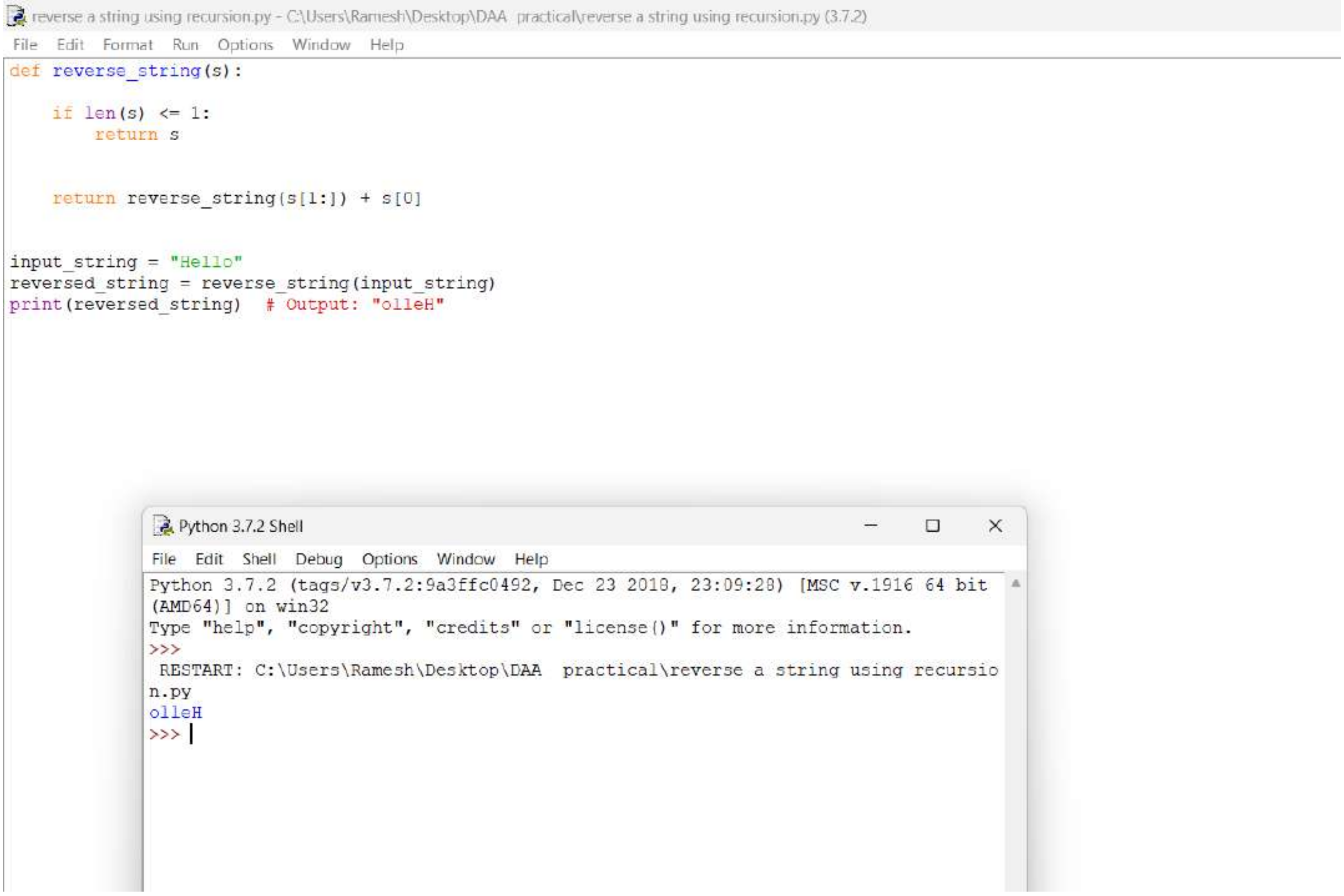
```
def generate_primes(n, primes=None, start=2):
    if primes is None:
        primes = []

    if start > n:
        return primes

    for i in range(start, n+1):
        is_prime = True
        for j in primes:
            if i % j == 0:
                is_prime = False
                break
        if is_prime:
            primes.append(i)

    return generate_primes(n, primes, primes[-1]+1)

limit = 50
prime_numbers = generate_primes(limit)
print("Prime numbers up to {limit}: {prime_numbers}")
```


reverse a string using recursion.py - C:\Users\Ramesh\Desktop\DAA practical\reverse a string using recursion.py (3.7.2)
File Edit Format Run Options Window Help
Python 3.7.2 Shell
File Edit Shell Debug Options Window Help

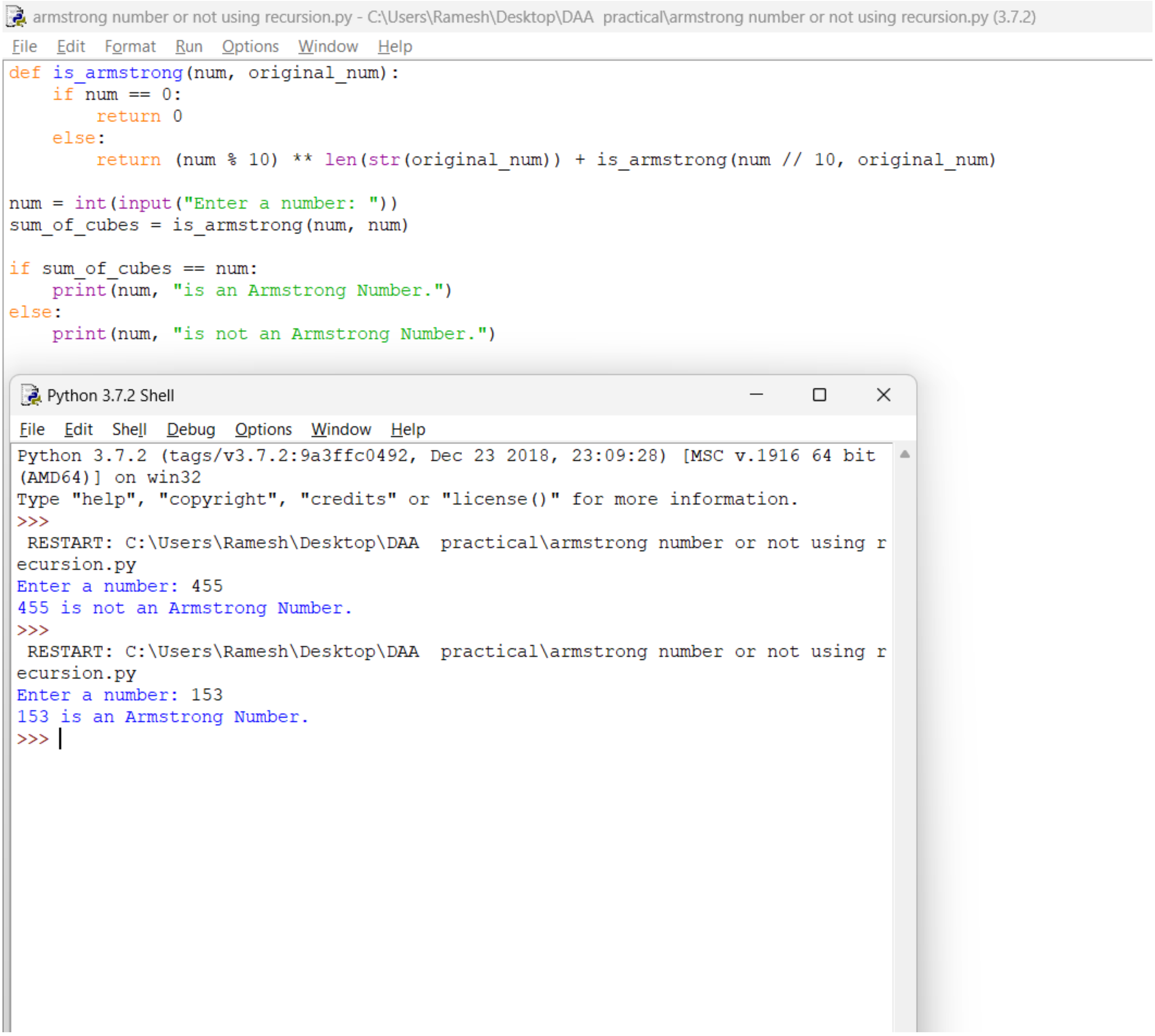
```python
def reverse_string(s):

    if len(s) <= 1:
        return s


    return reverse_string(s[1:]) + s[0]


input_string = "Hello"
reversed_string = reverse_string(input_string)
print(reversed_string)  # Output: "olleH"
```

```
Python 3.7.2 (tags/v3.7.2:9a3ffc0492, Dec 23 2018, 23:09:28) [MSC v.1916 64 bit
(AMD64)] on win32
Type "help", "copyright", "credits" or "license()" for more information.
>>>
 RESTART: C:\Users\Ramesh\Desktop\DAA  practical\reverse a string using recursio
n.py
olleH
>>> 
```


armstrong number or not using recursion.py - C:\Users\Ramesh\Desktop\DAA practical\armstrong number or not using recursion.py (3.7.2)
File Edit Format Run Options Window Help

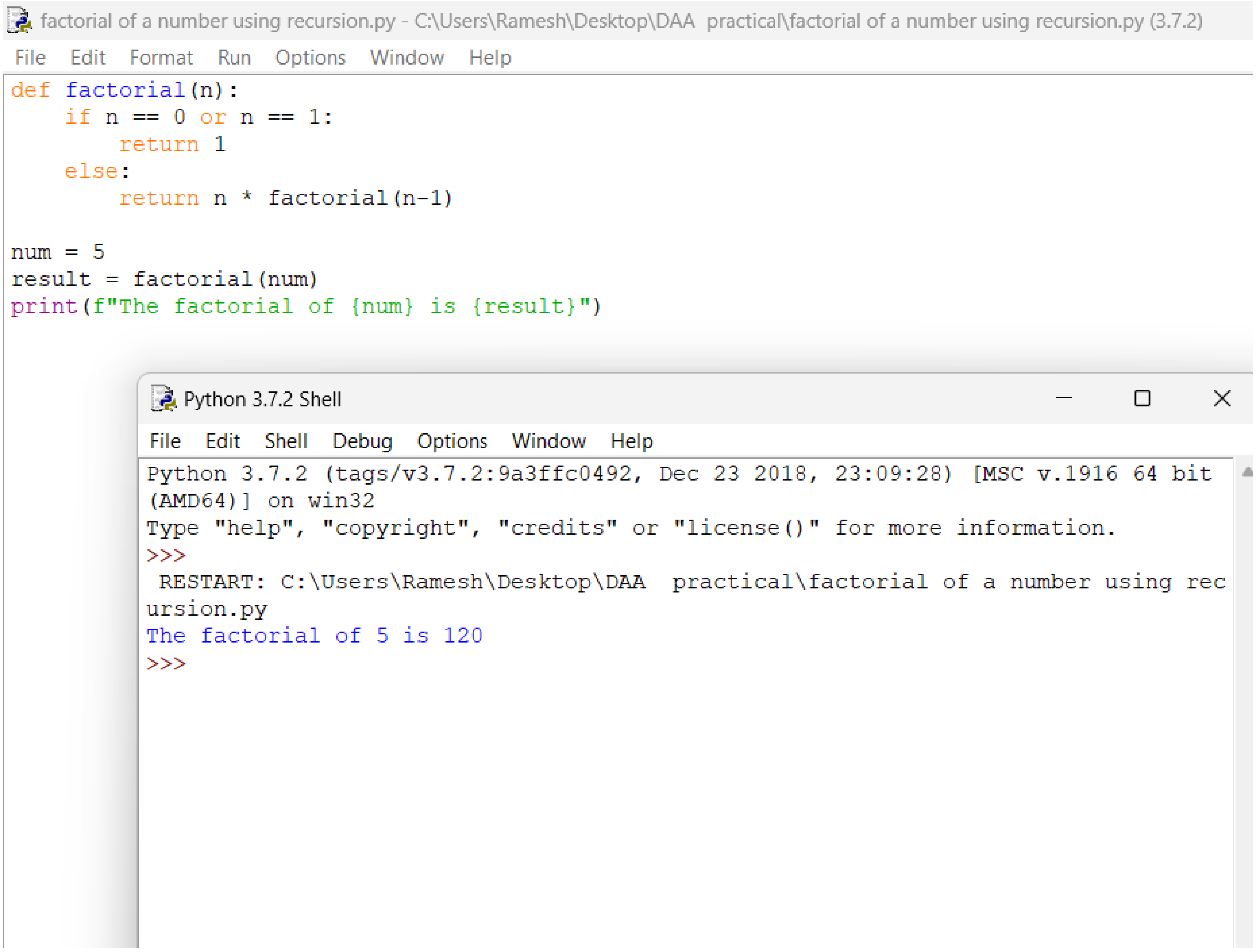
```
def is_armstrong(num, original_num):
    if num == 0:
        return 0
    else:
        return (num % 10) ** len(str(original_num)) + is_armstrong(num // 10, original_num)

num = int(input("Enter a number: "))
sum_of_cubes = is_armstrong(num, num)

if sum_of_cubes == num:
    print(num, "is an Armstrong Number.")
else:
    print(num, "is not an Armstrong Number.")
```

Python 3.7.2 Shell
File Edit Shell Debug Options Window Help

```
Python 3.7.2 (tags/v3.7.2:9a3ffc0492, Dec 23 2018, 23:09:28) [MSC v.1916 64 bit
(AMD64)] on win32
Type "help", "copyright", "credits" or "license()" for more information.
>>>
 RESTART: C:\Users\Ramesh\Desktop\DAA  practical\armstrong number or not using r
ecursion.py
Enter a number: 455
455 is not an Armstrong Number.
>>>
 RESTART: C:\Users\Ramesh\Desktop\DAA  practical\armstrong number or not using r
ecursion.py
Enter a number: 153
153 is an Armstrong Number.
>>> 
```



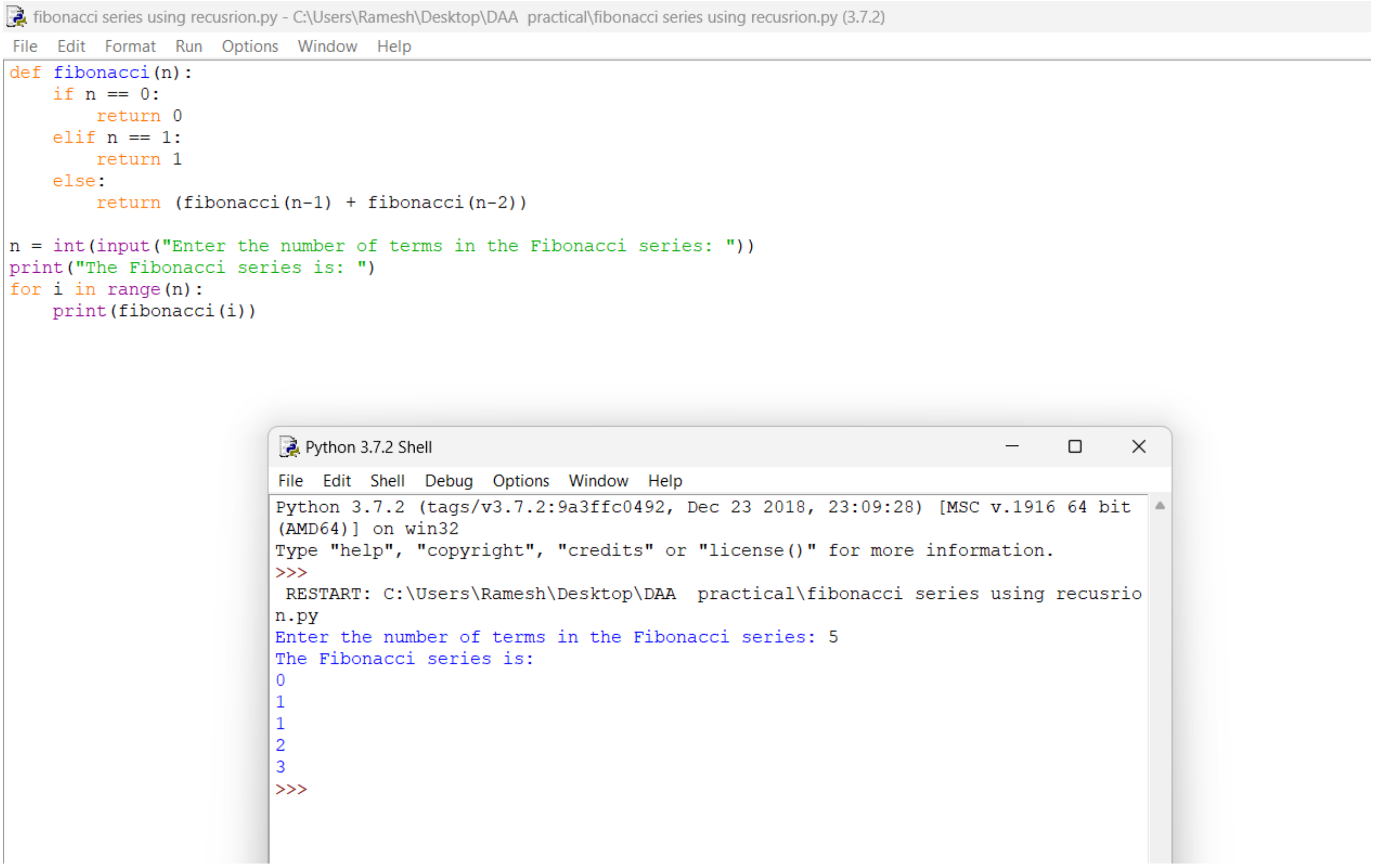
factorial of a number using recursion.py - C:\Users\Ramesh\Desktop\DAA practical\factorial of a number using recursion.py (3.7.2)
File Edit Format Run Options Window Help

```
def factorial(n):
    if n == 0 or n == 1:
        return 1
    else:
        return n * factorial(n-1)

num = 5
result = factorial(num)
print(f"The factorial of {num} is {result}")
```

Python 3.7.2 Shell
File Edit Shell Debug Options Window Help

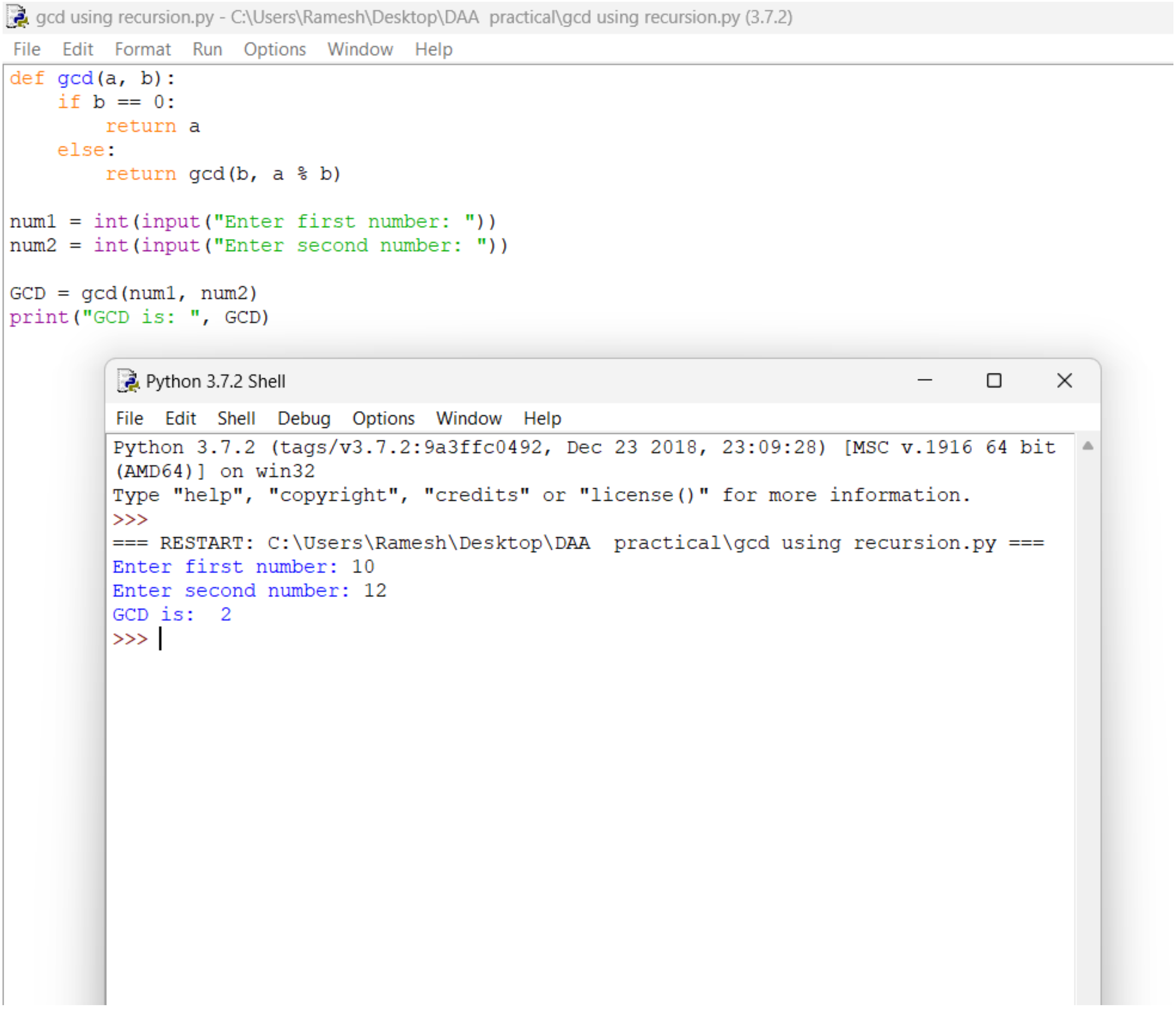
```
Python 3.7.2 (tags/v3.7.2:9a3ffc0492, Dec 23 2018, 23:09:28) [MSC v.1916 64 bit
(AMD64)] on win32
Type "help", "copyright", "credits" or "license()" for more information.
>>>
 RESTART: C:\Users\Ramesh\Desktop\DAA  practical\factorial of a number using rec
ursion.py
The factorial of 5 is 120
>>>
```


fibonacci series using recusrion.py - C:\Users\Ramesh\Desktop\DAA practical\fibonacci series using recusrion.py (3.7.2)
File Edit Format Run Options Window Help
Python 3.7.2 Shell
File Edit Shell Debug Options Window Help

```
def fibonacci(n):
    if n == 0:
        return 0
    elif n == 1:
        return 1
    else:
        return (fibonacci(n-1) + fibonacci(n-2))

n = int(input("Enter the number of terms in the Fibonacci series: "))
print("The Fibonacci series is: ")
for i in range(n):
    print(fibonacci(i))
```

```
Python 3.7.2 (tags/v3.7.2:9a3ffc0492, Dec 23 2018, 23:09:28) [MSC v.1916 64 bit
(AMD64)] on win32
Type "help", "copyright", "credits" or "license()" for more information.
>>>
 RESTART: C:\Users\Ramesh\Desktop\DAA  practical\fibonacci series using recusrio
n.py
Enter the number of terms in the Fibonacci series: 5
The Fibonacci series is:
0
1
1
2
3
>>>
```

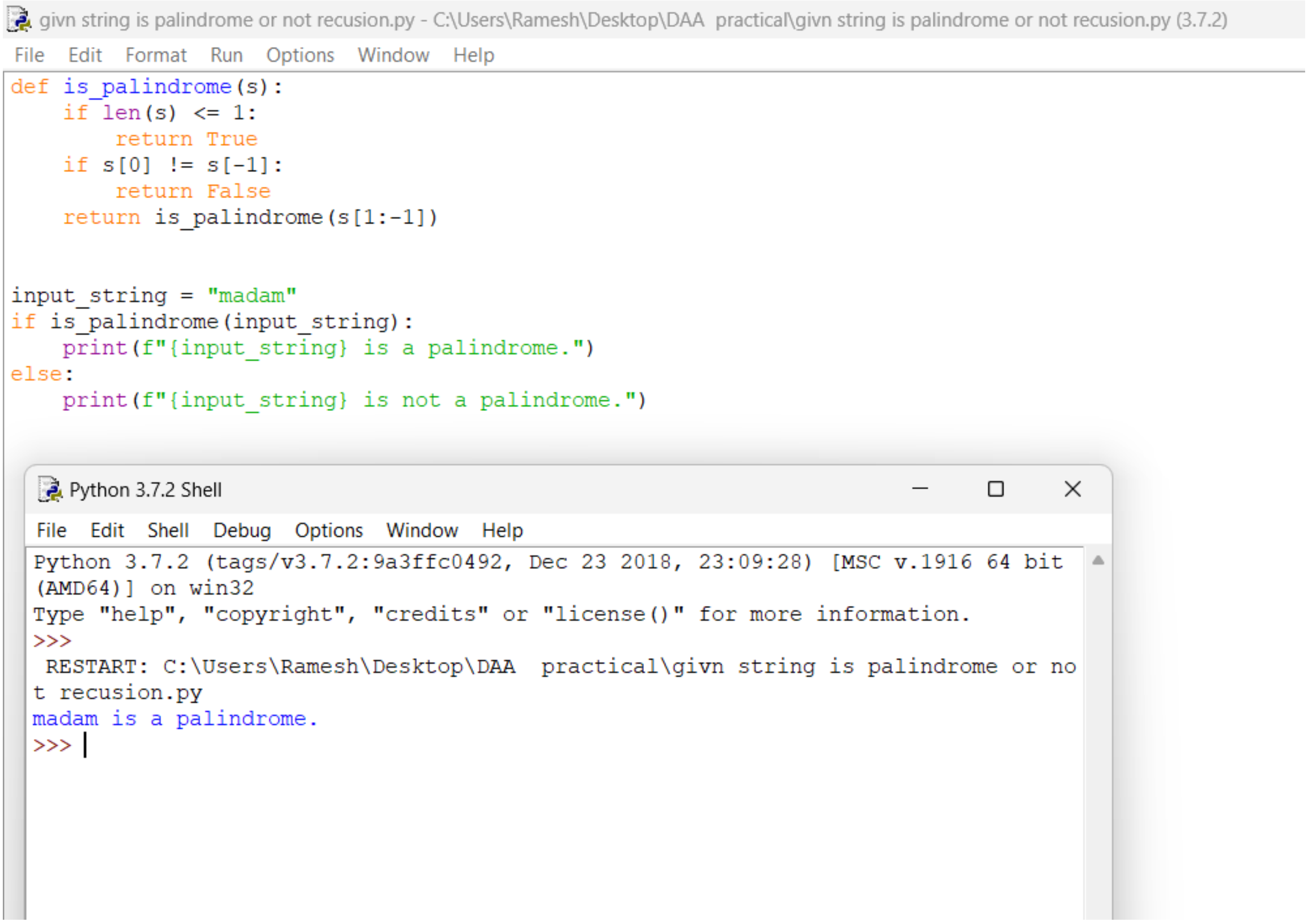

gcd using recursion.py - C:\Users\Ramesh\Desktop\DAA practical\gcd using recursion.py (3.7.2)
File Edit Format Run Options Window Help
Python 3.7.2 Shell
File Edit Shell Debug Options Window Help

```python
def gcd(a, b):
    if b == 0:
        return a
    else:
        return gcd(b, a % b)

num1 = int(input("Enter first number: "))
num2 = int(input("Enter second number: "))

GCD = gcd(num1, num2)
print("GCD is: ", GCD)
```

```
Python 3.7.2 (tags/v3.7.2:9a3ffc0492, Dec 23 2018, 23:09:28) [MSC v.1916 64 bit
(AMD64)] on win32
Type "help", "copyright", "credits" or "license()" for more information.
>>> 
=== RESTART: C:\Users\Ramesh\Desktop\DAA  practical\gcd using recursion.py ===
Enter first number: 10
Enter second number: 12
GCD is:  2
>>> 
```


givn string is palindrome or not recusion.py - C:\Users\Ramesh\Desktop\DAA practical\givn string is palindrome or not recusion.py (3.7.2)
File Edit Format Run Options Window Help

```python
def is_palindrome(s):
    if len(s) <= 1:
        return True
    if s[0] != s[-1]:
        return False
    return is_palindrome(s[1:-1])


input_string = "madam"
if is_palindrome(input_string):
    print(f"{input_string} is a palindrome.")
else:
    print(f"{input_string} is not a palindrome.")
```

Python 3.7.2 Shell
File Edit Shell Debug Options Window Help

```
Python 3.7.2 (tags/v3.7.2:9a3ffc0492, Dec 23 2018, 23:09:28) [MSC v.1916 64 bit
(AMD64)] on win32
Type "help", "copyright", "credits" or "license()" for more information.
>>>
 RESTART: C:\Users\Ramesh\Desktop\DAA  practical\givn string is palindrome or no
t recusion.py
madam is a palindrome.
>>> 
```