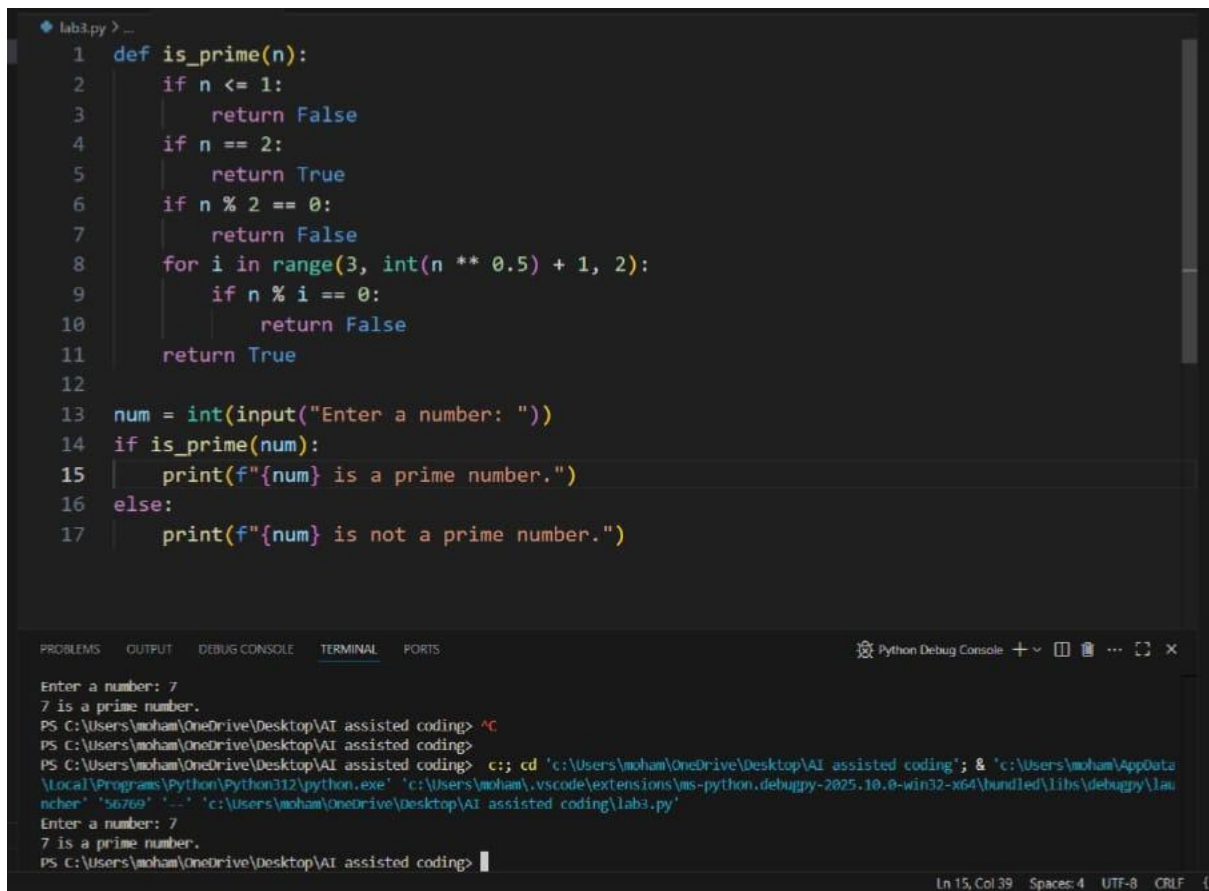


Generate a python function to check if a number is prime or not:



```
lab3.py > ...
1 def is_prime(n):
2     if n <= 1:
3         return False
4     if n == 2:
5         return True
6     if n % 2 == 0:
7         return False
8     for i in range(3, int(n ** 0.5) + 1, 2):
9         if n % i == 0:
10            return False
11    return True
12
13 num = int(input("Enter a number: "))
14 if is_prime(num):
15     print(f"{num} is a prime number.")
16 else:
17     print(f"{num} is not a prime number.")
```

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS Python Debug Console

```
Enter a number: 7
7 is a prime number.
PS C:\Users\moham\OneDrive\Desktop\AI assisted coding> ^C
PS C:\Users\moham\OneDrive\Desktop\AI assisted coding>
PS C:\Users\moham\OneDrive\Desktop\AI assisted coding> c:: cd 'c:\Users\moham\OneDrive\Desktop\AI assisted coding'; & 'c:\Users\moham\AppData
\Local\Programs\Python\Python312\python.exe' 'c:\Users\moham\.vscode\extensions\ms-python.debugpy-2025.10.0-win32-x64\bundled\libs\debugpy\lau
ncher' '56769' '...' 'c:\Users\moham\OneDrive\Desktop\AI assisted coding\lab3.py'
Enter a number: 7
7 is a prime number.
PS C:\Users\moham\OneDrive\Desktop\AI assisted coding>
```

Ln 15, Col 39 Spaces: 4 UTF-8 CRLF

Write a comment like function to reverse the string and use copilot to generate the function:

```
reverse_string.py > ...
1 def reverse_string():
2     s = input("Enter a string: ")
3     print(s[::-1])
4
5 # Example usage
6 reverse_string()
```

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS

```
Enter a string: 5
5
PS C:\Users\moham\OneDrive\Desktop\AI assisted coding> ^C
PS C:\Users\moham\OneDrive\Desktop\AI assisted coding>
PS C:\Users\moham\OneDrive\Desktop\AI assisted coding> c:: cd 'c:\Users\moham\OneDrive\Desktop\AI assisted coding'; & 'c:\User
s\moham\AppData\Local\Programs\Python\Python312\python.exe' 'c:\Users\moham\.vscode\extensions\ms-python.debugpy-2025.10.0-win3
2-x64\bundled\libs\debugpy\launcher' '50480' '--' 'c:\Users\moham\OneDrive\Desktop\AI assisted coding\reverse_string.py'
Enter a string: asdfg
gfdsa
PS C:\Users\moham\OneDrive\Desktop\AI assisted coding> []
```

+ ... [] x

powershell
Python Deb...

Ln 6, Col 17 Spaces: 4 UTF-8 CRLF {} P

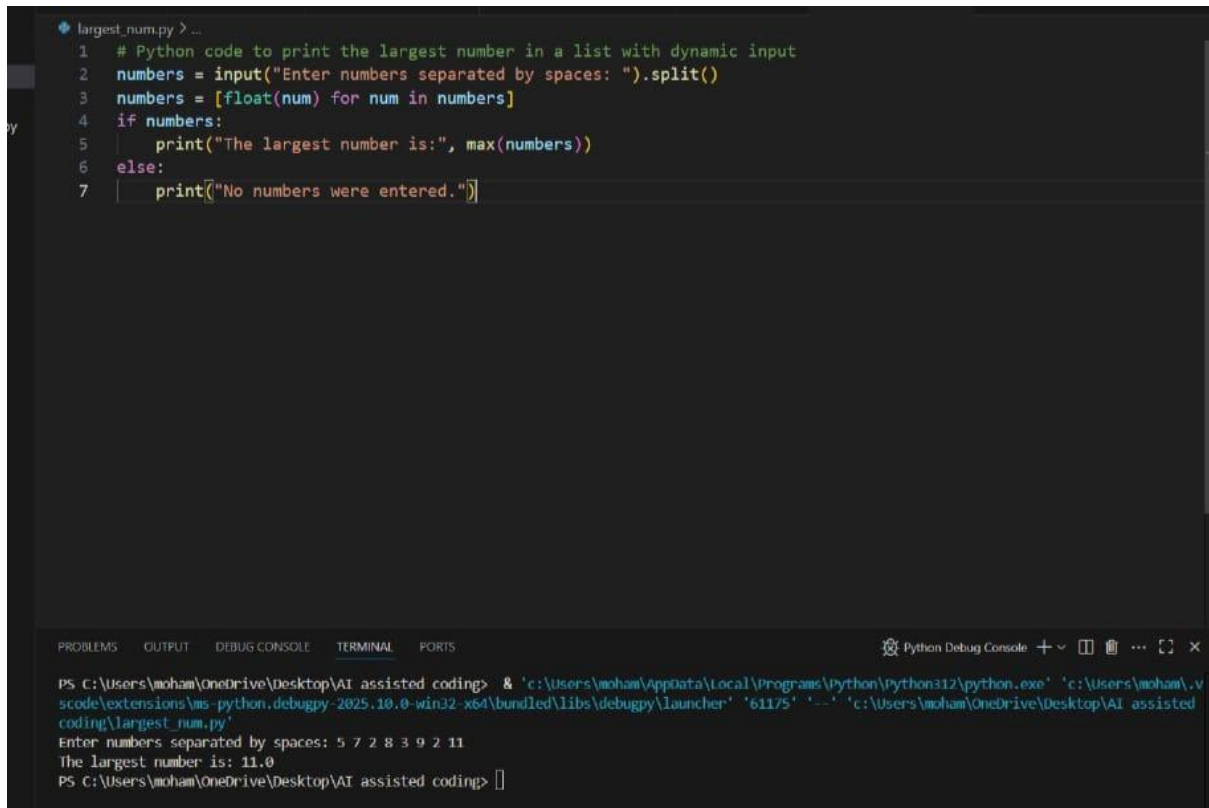
Generate both recursive and iterative versions of a factorial function using comments

```
rec_it.py > ...
1 def factorial_recursive(n):
2     if n <= 1:
3         return 1
4     return n * factorial_recursive(n - 1)
5
6 def factorial_iterative(n):
7     result = 1
8     for i in range(2, n + 1):
9         result *= i
10    return result
11
12 num = int(input("Enter a non-negative integer: "))
13 print("Recursive:", factorial_recursive(num))
14 print("Iterative:", factorial_iterative(num))
15
```

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS Python Debug Console + - [] ... [] x

```
PS C:\Users\moham\OneDrive\Desktop\AI assisted coding> & 'c:\Users\moham\AppData\Local\Programs\Python\Python312\python.exe' 'c:\Users\moham\
.vscode\extensions\ms-python.debugpy-2025.10.0-win32-x64\bundle\libs\debugpy\launcher' '60339' '-' 'c:\Users\moham\OneDrive\Desktop\AI assis
ted coding\rec_it.py'
Enter a non-negative integer: 5
Recursive: 120
Iterative: 120
PS C:\Users\moham\OneDrive\Desktop\AI assisted coding> |
```

Use copilot to find the largest number in a list .Access code quality and efficiency.



The image shows a Visual Studio Code editor window with a Python file named `largest_num.py`. The code is as follows:

```
1 # Python code to print the largest number in a list with dynamic input
2 numbers = input("Enter numbers separated by spaces: ").split()
3 numbers = [float(num) for num in numbers]
4 if numbers:
5     print("The largest number is:", max(numbers))
6 else:
7     print("No numbers were entered.")
```

Below the editor, the `TERMINAL` panel is active, showing the command prompt output:

```
PS C:\Users\moham\OneDrive\Desktop\AI assisted coding> & 'c:\Users\moham\AppData\Local\Programs\Python\Python312\python.exe' 'c:\Users\moham\vscode\extensions\ms-python.debugpy-2025.10.0-win32-x64\bundle\libs\debugpy\launcher' '61175' '--' 'c:\Users\moham\OneDrive\Desktop\AI assisted coding\largest_num.py'
Enter numbers separated by spaces: 5 7 2 8 3 9 2 11
The largest number is: 11.0
PS C:\Users\moham\OneDrive\Desktop\AI assisted coding>
```