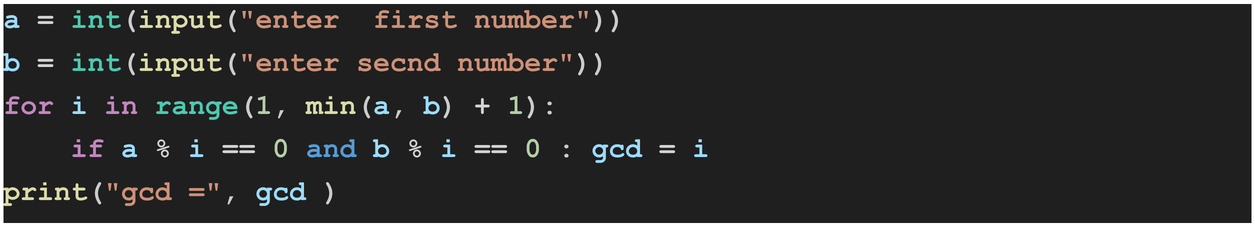
ASSIGNMENT 1

Submitted to :- Ms. Anit James
Submitted by:-Fable K Lonappan
Roll no : 24
S9 INT MCA

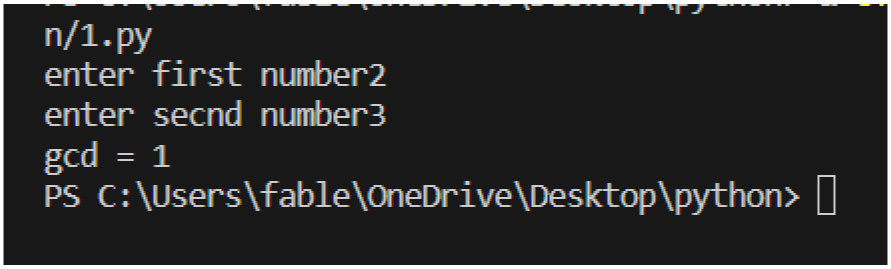
1 . Program to find the GCD of 2 numbers

Code.



```
a = int(input("enter  first number"))
b = int(input("enter secnd number"))
for i in range(1, min(a, b) + 1):
    if a % i == 0 and b % i == 0 : gcd = i
print("gcd =", gcd )
```

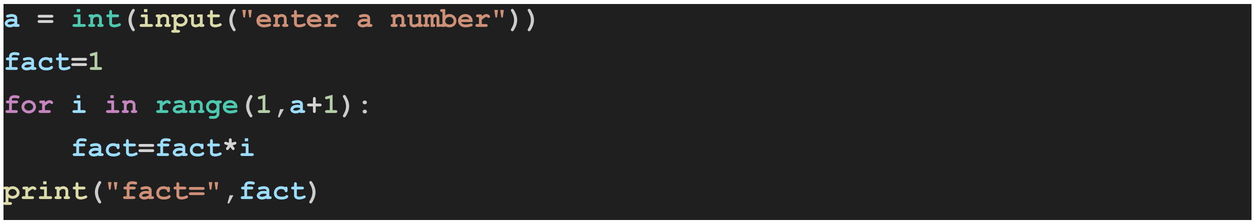
Output:



```
n/1.py
enter first number2
enter secnd number3
gcd = 1
PS C:\Users\fable\OneDrive\Desktop\python> 
```

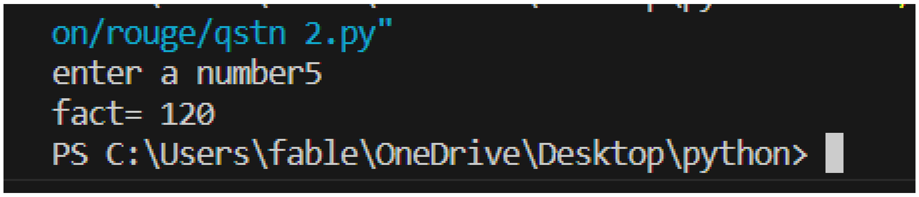
2.Program to find the factorial of a number

Code:



```
a = int(input("enter a number"))
fact=1
for i in range(1,a+1):
    fact=fact*i
print("fact=",fact)
```

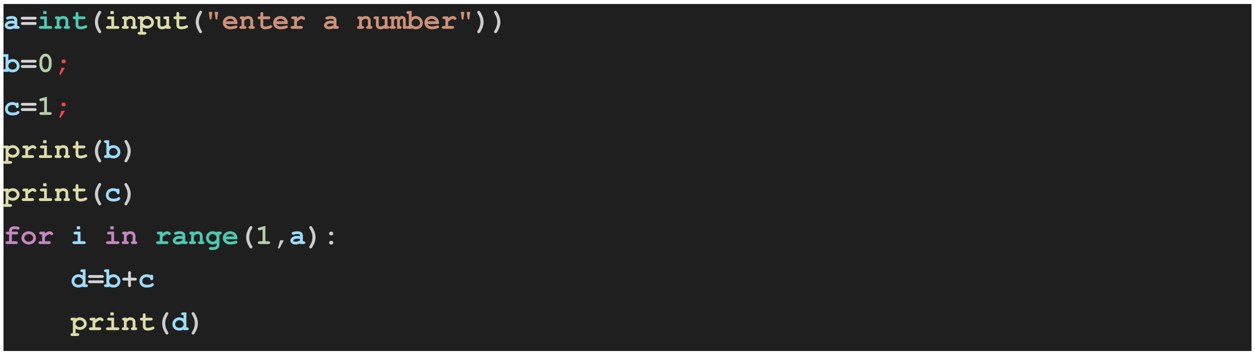
Output:



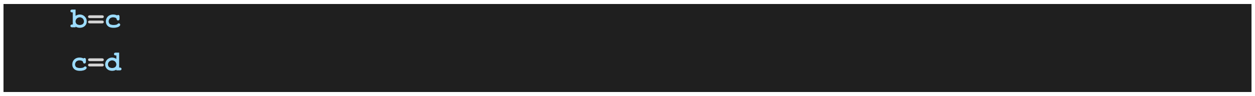
```
on/rouge/qstn 2.py"
enter a number5
fact= 120
PS C:\Users\fable\OneDrive\Desktop\python> 
```

3.program to generate fibonacci series of N terms

Code:

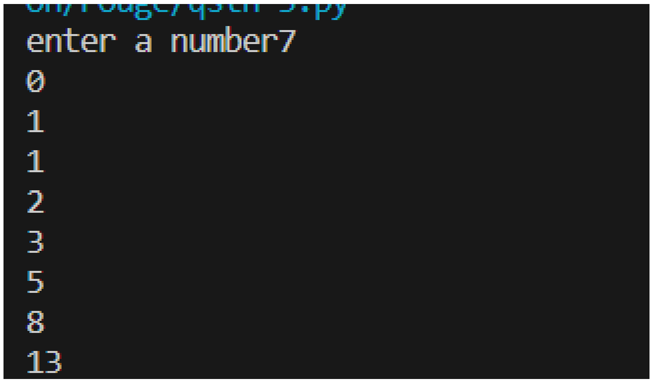


```
a=int(input("enter a number"))
b=0;
c=1;
print(b)
print(c)
for i in range(1,a):
    d=b+c
    print(d)
```



```
    b=c
    c=d
```

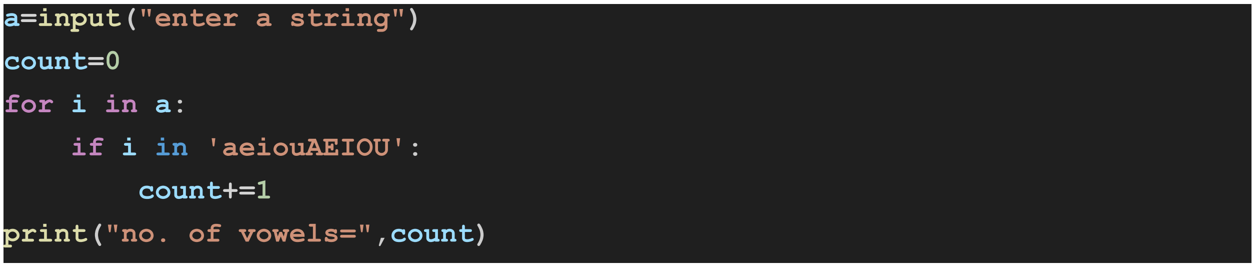
Output:



```
on/rouge/qstn 3.py
enter a number7
0
1
1
2
3
5
8
13
```

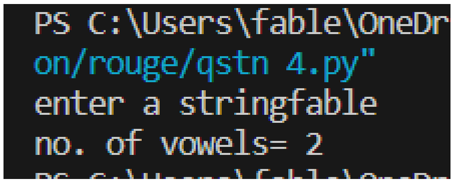
4.program to count the number of vowels

Code:



```
a=input("enter a string")
count=0
for i in a:
    if i in 'aeiouAEIOU':
        count+=1
print("no. of vowels=",count)
```

Output:



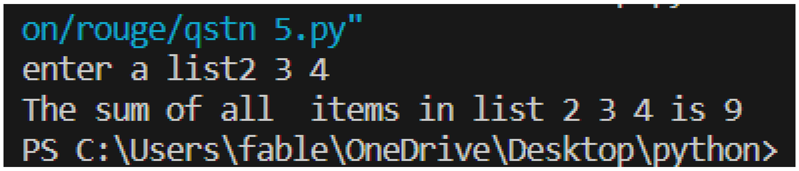
```
PS C:\Users\fable\OneDr
on/rouge/qstn 4.py"
enter a stringfable
no. of vowels= 2
PS C:\Users\fable\OneDr
```

5.Program to find the sum of all items in a list

Code:

list=input("enter a list")
list1=map(int,list.split())
sum=0
for i in list1:
sum+=i
print("The sum of all items in list",list,"is",sum)

Output:



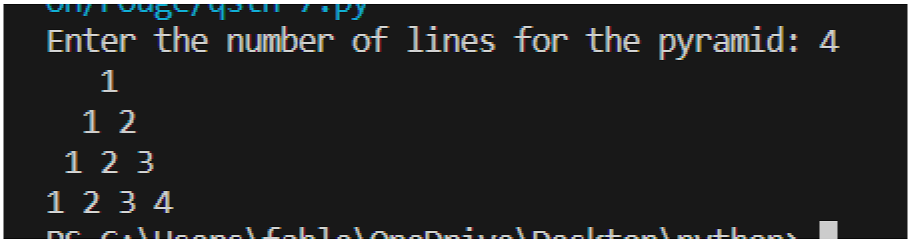
```
on/rouge/qstn 5.py"
enter a list2 3 4
The sum of all  items in list 2 3 4 is 9
PS C:\Users\fable\OneDrive\Desktop\python>
```

7.write a program that prints the following pyramid on the screen . The number of lines must be obtained from the user as input.

Code:

```
def print_pyramid(n):
for i in range(1, n + 1):
print(" " * (n - i), end="")
for j in range(1, i + 1):
print(j, end=" ")
print()
n = int(input("Enter the number of lines for the pyramid: "))
print_pyramid(n)
```

Output:



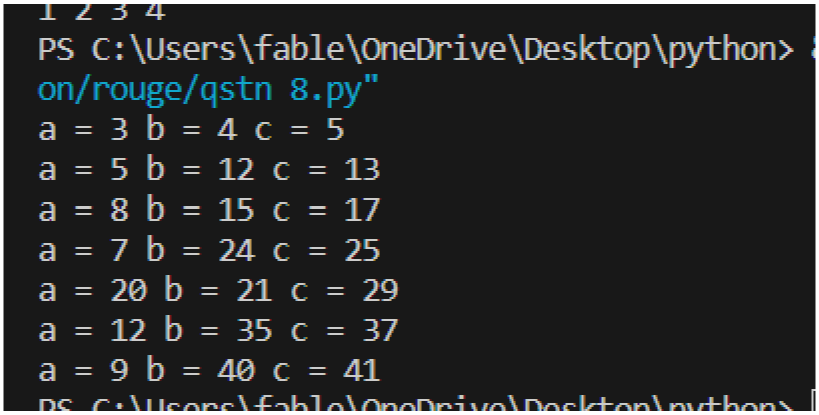
```
on/rouge/qstn 7.py
Enter the number of lines for the pyramid: 4
   1
  1 2
 1 2 3
1 2 3 4
PS C:\Users\fable\OneDrive\Desktop\python>
```

8.write a python program to find primitive pythagorean traids a pythagorean traid has the property a2+b2=c2. by primitive we mean traids that do not 'depend' on others.for example
(4,3,5) is a variants of (3,4,5) and hence is not primitive .And (10,24,26) is easily derived from
(5,12,13) and should not be displayed by our program. write a program to print primitive pythagorean traid.the program should generate all traids a,b values in the range 0-50
Code:

```
for i in range(1, 50):
for j in range(1, i):
for k in range(1, j):
if k * k + j * j == i * i:
flag = 0
for l in range(2, i):
if i % l == 0 and j % l == 0 and k % l == 0:
```

```
flag = 1
break
if flag:
continue
print("a =", k, "b =", j, "c =", i)
```

Output:



```
1 2 3 4
PS C:\Users\fable\OneDrive\Desktop\python>
on/rouge/qstn 8.py"
a = 3 b = 4 c = 5
a = 5 b = 12 c = 13
a = 8 b = 15 c = 17
a = 7 b = 24 c = 25
a = 20 b = 21 c = 29
a = 12 b = 35 c = 37
a = 9 b = 40 c = 41
PS C:\Users\fable\OneDrive\Desktop\python>
```

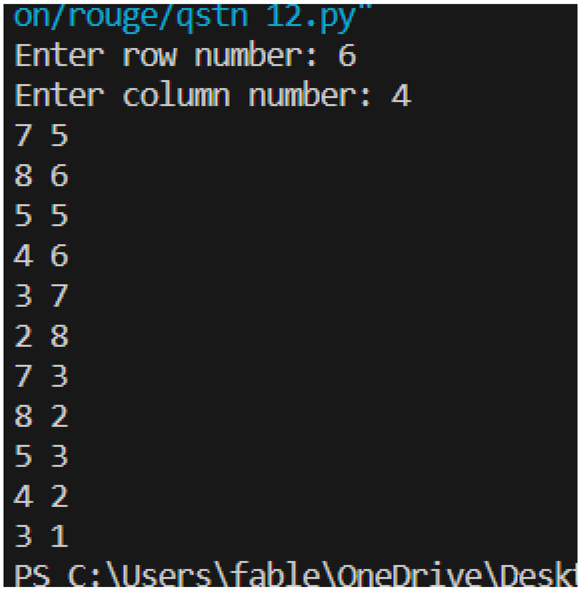
12. Given an empty chessboard and one bishop placed in any square,say(r,c),generate the list
of all squares the bishop could move to

Code:

```
r = int(input("Enter row number: "))
c = int(input("Enter column number: "))
if r in range(1, 9) and c in range(1, 9):
print("Possible Movements in (row, col):")
# Diagonal movement
for i in range(1, 9):
if r + i <= 8 and c + i <= 8:
print(r + i, c + i)
else:
break
for i in range(1, 9):
if r - i >= 1 and c + i <= 8:
print(r - i, c + i)
else:
break
for i in range(1, 9):
if r + i <= 8 and c - i >= 1:
print(r + i, c - i)
else:
break
```

```
for i in range(1, 9):
if r - i >= 1 and c - i >= 1:
print(r - i, c - i)
else:
break
else:
print("Invalid range for row or column")
```

Output:



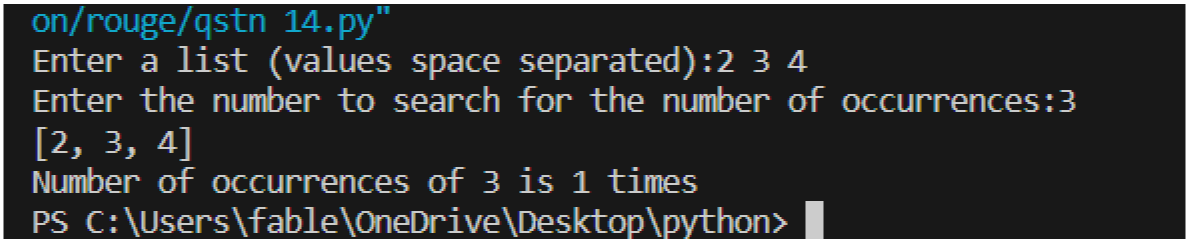
```
on/rouge/qstn 12.py"
Enter row number: 6
Enter column number: 4
7 5
8 6
5 5
4 6
3 7
2 8
7 3
8 2
5 3
4 2
3 1
PS C:\Users\fable\OneDrive\Deskt
```

14.Write a python program to count in a given list

Code:

```
str_input = input("Enter a list (values space separated):")
lis = list(map(int, str_input.split()))
n = int(input("Enter the number to search for the number of occurrences:"))
print(lis)
print("Number of occurrences of", n, "is", lis.count(n), "times")
```

Output:



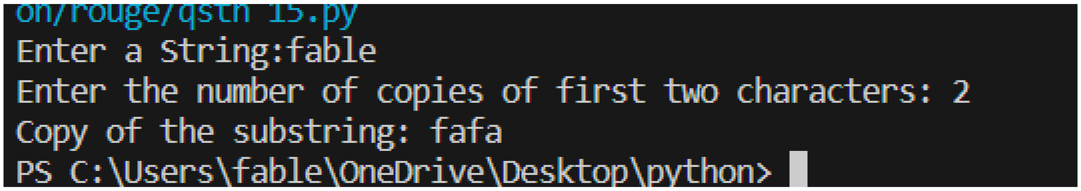
```
on/rouge/qstn 14.py"
Enter a list (values space separated):2 3 4
Enter the number to search for the number of occurrences:3
[2, 3, 4]
Number of occurrences of 3 is 1 times
PS C:\Users\fable\OneDrive\Desktop\python>
```

15 . write a python program to get the n (non negative integer) copies of the first 2 characters
of a given string . print n copies of the whole string if the length is less than 2

Code:

```
str_input = input("Enter a String:")
n = int(input("Enter the number of copies of first two characters: "))
if len(str_input) < 2:
flen = len(str_input)
else:
flen = 2
substr = str_input[:flen]
result = substr * n
print("Copy of the substring:", result)
```

Output:



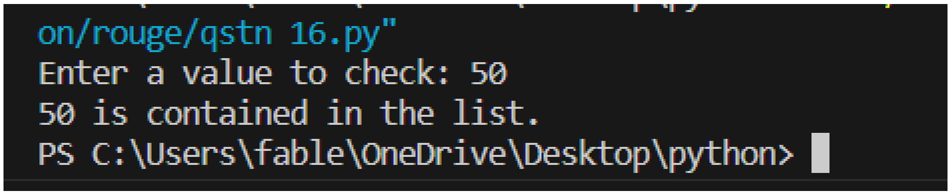
```
on/rouge/qstn 15.py
Enter a String:fable
Enter the number of copies of first two characters: 2
Copy of the substring: fafa
PS C:\Users\fable\OneDrive\Desktop\python>
```

16 . write a python program to check whether a specified value is contained value is contained in a group of values

Code:

```
values = [10, 20, 30, 40, 50]
value_to_check = int(input("Enter a value to check: "))
if value_to_check in values:
print(f"{value_to_check} is contained in the list.")
else:
print(f"{value_to_check} is not contained in the list.")
```

Output:



```
on/rouge/qstn 16.py"
Enter a value to check: 50
50 is contained in the list.
PS C:\Users\fable\OneDrive\Desktop\python>
```

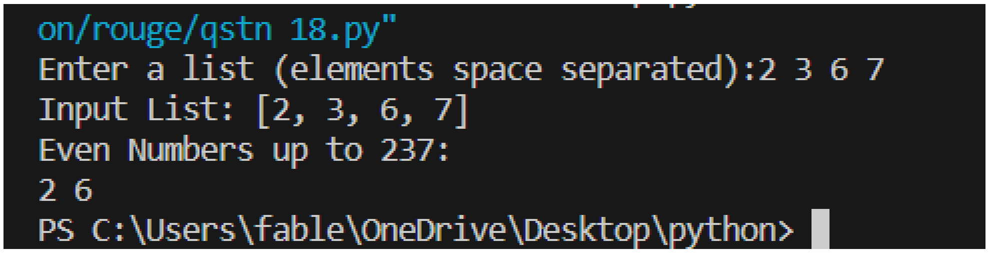
18 . write a python program to print all even numbers from a given numbers list in the same order and stop the printing if any numbers that come after 237 in the sequence

Code:

```
lis = input("Enter a list (elements space separated):")
lis1 = list(map(int, lis.split()))
print("Input List:", lis1)
print("Even Numbers up to 237:")
for x in lis1:
```

```
if x == 237:
break
elif x % 2 == 0:
print(x, end=' ')
```

Output:



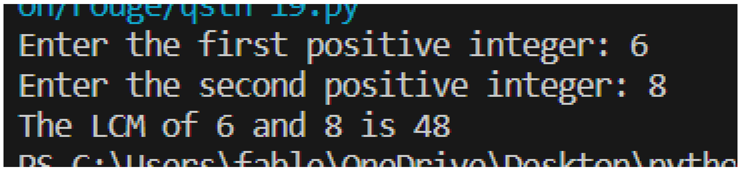
```
on/rouge/qstn 18.py"
Enter a list (elements space separated):2 3 6 7
Input List: [2, 3, 6, 7]
Even Numbers up to 237:
2 6
PS C:\Users\fable\OneDrive\Desktop\python>
```

19 . write a python program to get the least common multiple of two positive integers

Code:

```
def lcm(a, b):
return (a * b)
num1 = int(input("Enter the first positive integer: "))
num2 = int(input("Enter the second positive integer: "))
if num1 <= 0 or num2 <= 0:
print("Please enter positive integers.")
else:
result = lcm(num1, num2)
print(f"The LCM of {num1} and {num2} is {result}")
```

Output:



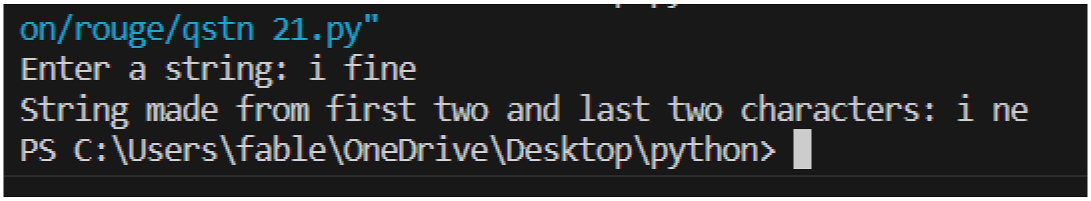
```
on/rouge/qstn 19.py
Enter the first positive integer: 6
Enter the second positive integer: 8
The LCM of 6 and 8 is 48
PS C:\Users\fable\OneDrive\Desktop\pytho
```

21 . write a python program to get a string made of the first 2 and the last 2 chars from a given a string. if the string length is less than 2, return instead the empty string

Code:

```
str1 = input("Enter a string: ")
if len(str1) < 2:
print("")
else:
result = str1[0:2] + str1[-2:]
print("String made from first two and last two characters:", result)
```

Output:



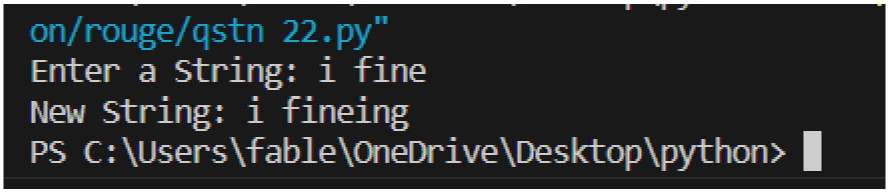
```
on/rouge/qstn 21.py"
Enter a string: i fine
String made from first two and last two characters: i ne
PS C:\Users\fable\OneDrive\Desktop\python>
```

22 . write a python program to add 'ing' at the end of a given string(length should be at least 3).if the given string is already ends with 'ing' then add 'ly' instead. if the string length of the given string is less than 3, leave it unchanged

Code:

```
strl = input("Enter a String: ")
if len(strl) >= 3:
if strl[-3:] == 'ing':
strl += 'ly'
else:
strl += 'ing'
print("New String:", strl)
```

Output:



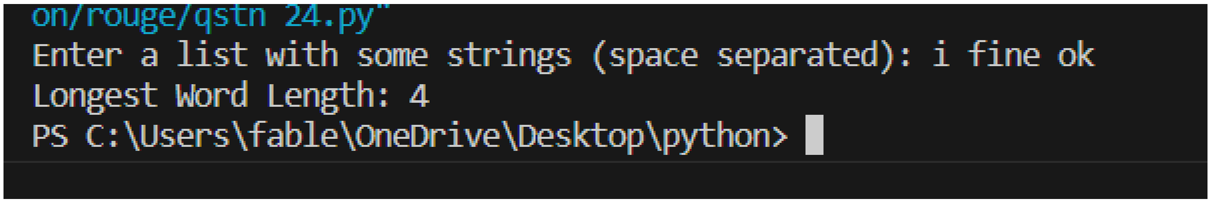
```
on/rouge/qstn 22.py"
Enter a String: i fine
New String: i fineing
PS C:\Users\fable\OneDrive\Desktop\python>
```

24 .write a python function that takes a list of words and return the length of the longest one

Code:

```
def find_longest_word_length(words_list):
word_len = []
for word in words_list:
word_len.append((len(word), word))
word_len.sort()
return word_len[-1][0] if word_len else 0
input_str = input("Enter a list with some strings (space separated): ")
words_list = input_str.split()
longest_word_length = find_longest_word_length(words_list)
print("Longest Word Length:", longest_word_length)
```

Output:



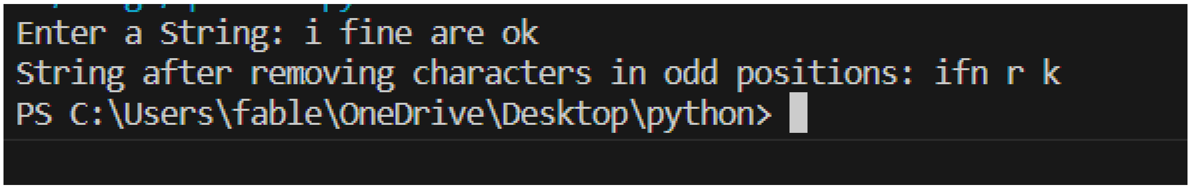
```
on/rouge/qstn 24.py"
Enter a list with some strings (space separated): i fine ok
Longest Word Length: 4
PS C:\Users\fable\OneDrive\Desktop\python>
```

25 . write a python program to remove the characters which have oddstring

Code:

```
strl = input("Enter a String: ")
result = strl[::2]
print("String after removing characters in odd positions:", result)
```

Output:



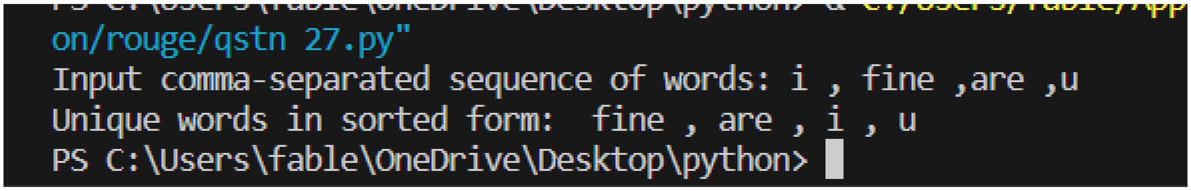
```
Enter a String: i fine are ok
String after removing characters in odd positions: ifn r k
PS C:\Users\fable\OneDrive\Desktop\python>
```

27 . write a python program that accepts a comma seperated sequence of words as input and prints the unique words in sorted form(alphanumerically)

Code:

```
items = input("Input comma-separated sequence of words: ")
words = items.split(",")
unique_words = sorted(set(words))
result = ", ".join(unique_words)
print("Unique words in sorted form: " + result)
```

Output:



```
on/rouge/qstn 27.py"
Input comma-separated sequence of words: i , fine ,are ,u
Unique words in sorted form:  fine , are , i , u
PS C:\Users\fable\OneDrive\Desktop\python>
```

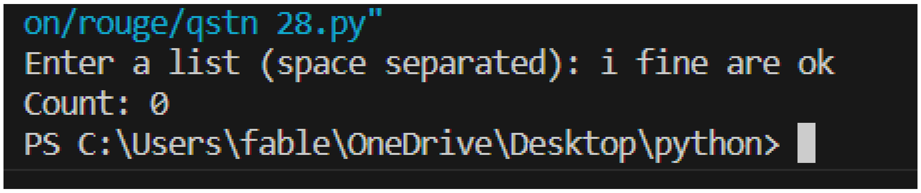
28 . Write a python program to count the number of strings where the string length is 2 or more and the first and last character are same from a given list of strings.

Code:

```
input_str = input("Enter a list (space separated): ")
words_list = input_str.split()
count = 0
```

```
for word in words_list:
if len(word) >= 2 and word[0] == word[-1]:
count += 1
print("Count:", count)
```

Output:



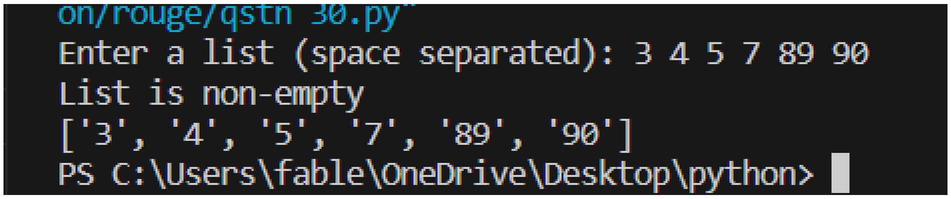
```
on/rouge/qstn 28.py"
Enter a list (space separated): i fine are ok
Count: 0
PS C:\Users\fable\OneDrive\Desktop\python>
```

30 . Write a python program to check a list is empty or not

Code:

```
input_str = input("Enter a list (space separated): ")
lis = input_str.split()
if not lis:
print("List is empty")
else:
print("List is non-empty")
print(lis)
```

Output:



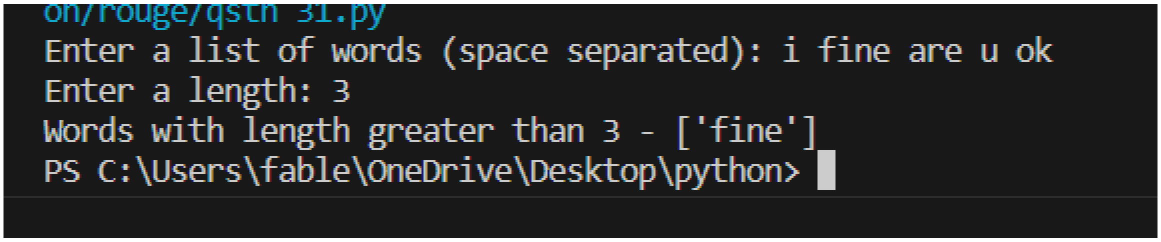
```
on/rouge/qstn 30.py"
Enter a list (space separated): 3 4 5 7 89 90
List is non-empty
['3', '4', '5', '7', '89', '90']
PS C:\Users\fable\OneDrive\Desktop\python>
```

31 . Write a python program to find the list of words that are longer than n from a given list of words

Code:

```
input_str = input("Enter a list of words (space separated): ")
length_to_check = int(input("Enter a length: "))
words_list = input_str.split()
word_len = []
for word in words_list:
if len(word) > length_to_check:
word_len.append(word)
print("Words with length greater than", length_to_check, "-", word_len)
```

Output:



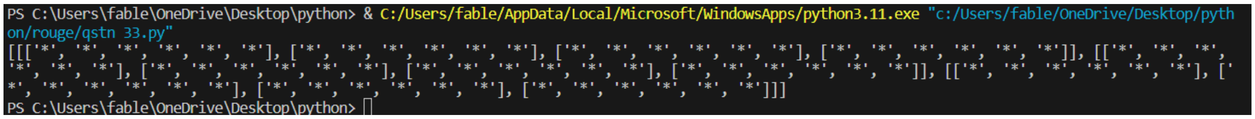
```
on/rouge/qstn 31.py
Enter a list of words (space separated): i fine are u ok
Enter a length: 3
Words with length greater than 3 - ['fine']
PS C:\Users\fable\OneDrive\Desktop\python> 
```

33. Write a python program to generate a 3*4*6 3D array whose each element is *.

Code:

```
array = [[['*' for col in range(6)] for col in range(4)] for row in range(3)]
print(array)
```

Output:



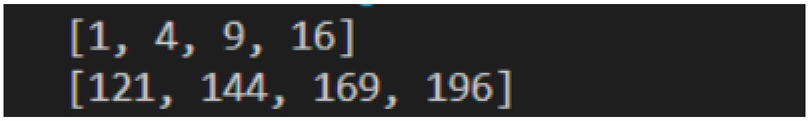
```
PS C:\Users\fable\OneDrive\Desktop\python> & C:/Users/fable/AppData/Local/Microsoft/WindowsApps/python3.11.exe "c:/Users/fable/OneDrive/Desktop/pyth
on/rouge/qstn 33.py"
[[['*', '*', '*', '*', '*', '*'], ['*', '*', '*', '*', '*', '*'], ['*', '*', '*', '*', '*', '*'], ['*', '*', '*', '*', '*', '*']], [['*', '*', '*',
'*', '*', '*'], ['*', '*', '*', '*', '*', '*'], ['*', '*', '*', '*', '*', '*'], ['*', '*', '*', '*', '*', '*']], [['*', '*', '*', '*', '*', '*'], ['
*', '*', '*', '*', '*', '*'], ['*', '*', '*', '*', '*', '*'], ['*', '*', '*', '*', '*', '*']]]
PS C:\Users\fable\OneDrive\Desktop\python> 
```

35 . Write a python program to generate and print a list of first and last 5 elements where the values are squares of numbers between 1 and 30 (both included).

Code:

```
l=list()
for i in range(1,15):
l.append(i**2)
print(l[:4])
print(l[-4:])
```

Output:



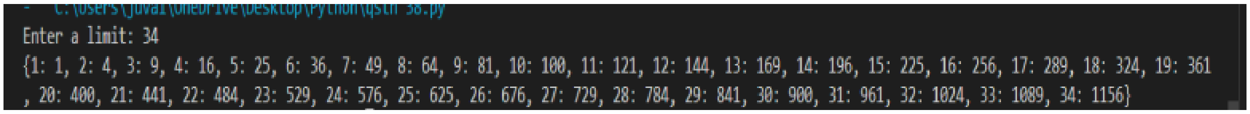
```
[1, 4, 9, 16]
[121, 144, 169, 196]
```

38 .Write a python script to generate and print a dictionary that contains number (between 1 and n) in the form (x*x*X)

Code:

```
n = int(input("Enter a limit: "))
d = dict()
for x in range(1, n + 1):
d[x] = x * x
print(d)
```

Output:



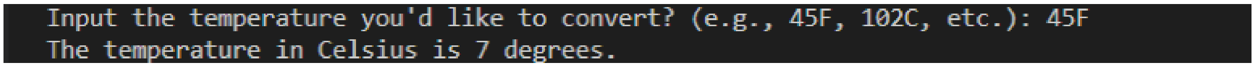
```
-   C:\Users\Juval\OneDrive\Desktop\Python\qstn 38.py
Enter a limit: 34
{1: 1, 2: 4, 3: 9, 4: 16, 5: 25, 6: 36, 7: 49, 8: 64, 9: 81, 10: 100, 11: 121, 12: 144, 13: 169, 14: 196, 15: 225, 16: 256, 17: 289, 18: 324, 19: 361
, 20: 400, 21: 441, 22: 484, 23: 529, 24: 576, 25: 625, 26: 676, 27: 729, 28: 784, 29: 841, 30: 900, 31: 961, 32: 1024, 33: 1089, 34: 1156}
```

41 . Write a python program to convert temperatures to and from celsius ,fahrenheit

Code:

```
temp = input("Input the temperature you'd like to convert? (e.g., 45F, 102C, etc.): ")
degree = int(temp[:-1])
input_convention = temp[-1]
if input_convention.upper() == "C":
result = int(round((9 * degree) / 5 + 32))
output_convention = "Fahrenheit"
elif input_convention.upper() == "F":
result = int(round((degree - 32) * 5 / 9))
output_convention = "Celsius"
else:
print("Input proper convention.")
exit()
print("The temperature in", output_convention, "is", result, "degrees.")
```

Output:



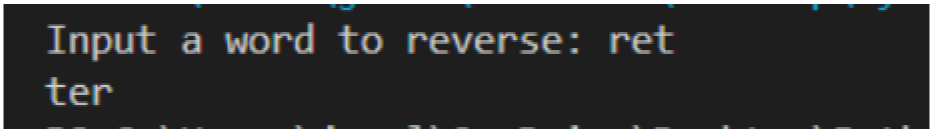
```
Input the temperature you'd like to convert? (e.g., 45F, 102C, etc.): 45F
The temperature in Celsius is 7 degrees.
```

43. Write a python program that accept a word from the user and reverse it

Code:

```
word = input("Input a word to reverse: ")
reversed_word = word[::-1]
print(reversed_word)
```

Output:



```
Input a word to reverse: ret
ter
```

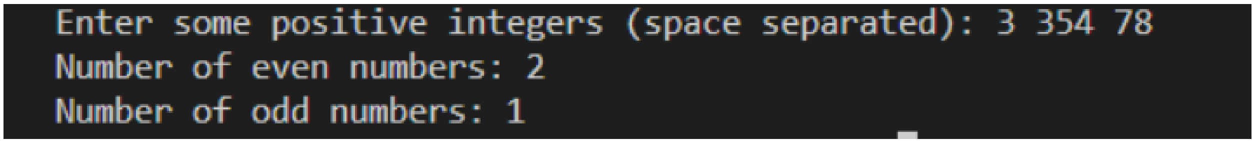
44 . Write a python program that counts odd and even numbers from a list

Code:

```
numbers_list = input("Enter some positive integers (space separated): ").split()
count_even = 0
count_odd = 0
for x in numbers_list:
x = int(x)
if x % 2 == 0:
count_even += 1
```

```
else:
count_odd += 1
print("Number of even numbers:", count_even)
print("Number of odd numbers:", count_odd)
```

Output:



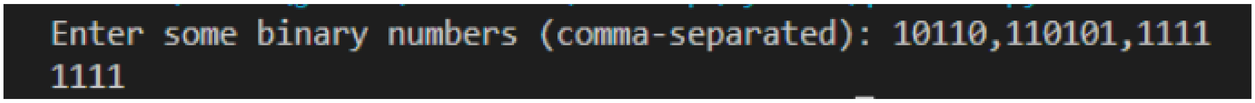
```
Enter some positive integers (space separated): 3 354 78
Number of even numbers: 2
Number of odd numbers: 1
```

47 . Write a python program which accepts a sequence of comma separated 4 digits binary numbers as its input and print the numbers that are divisible by 5 in a comma separated sequence

Code:

```
items = []
num_input = input("Enter some binary numbers (comma-separated): ")
num_list = num_input.split(',')
for p in num_list:
x = int(p, 2)
if x % 5 == 0:
items.append(p)
print(','.join(items))
```

Output:



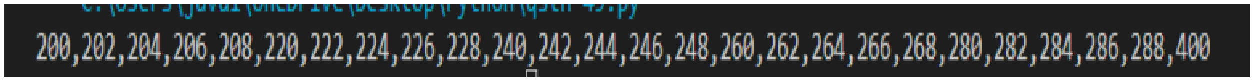
```
Enter some binary numbers (comma-separated): 10110,110101,1111
1111
```

49 . Write a python program to find numbers between 100 and 400 (both includes) where each digit of a number is an even number . The numbers obtained should be printed in a comma-separated sequence

Code:

```
items = []
for i in range(100, 401):
s = str(i)
if (int(s[0]) % 2 == 0) and (int(s[1]) % 2 == 0) and (int(s[2]) % 2 == 0):
items.append(s)
print(",".join(items))
```

Output:



```
200,202,204,206,208,220,222,224,226,228,240,242,244,246,248,260,262,264,266,268,280,282,284,286,288,400
```

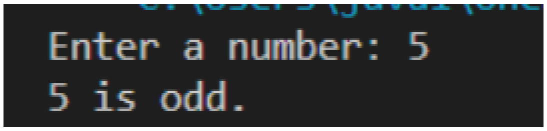
FUNCTIONS

1.Write a python function to check whether a number is even or odd

Code:

```
def is_even_or_odd():
number = int(input("Enter a number: "))
if number % 2 == 0:
print(f"{number} is even.")
else:
print(f"{number} is odd.")
is_even_or_odd()
```

Output:

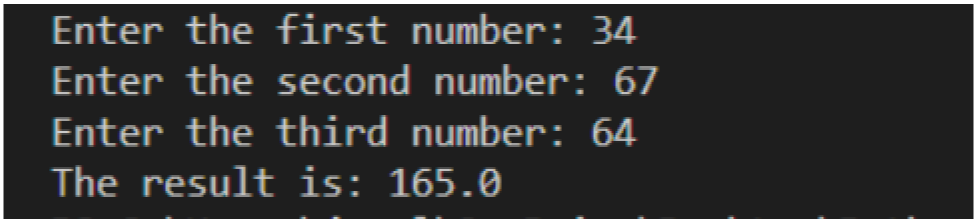


```
Enter a number: 5
5 is odd.
```

3.Write a python program to calculate the sum of three given numbers,if the values are equal then return thrice of their sum

```
def calculate_sum(num1, num2, num3):
total_sum = num1 + num2 + num3
if num1 == num2 == num3:
return total_sum * 3
else:
return total_sum
num1 = float(input("Enter the first number: "))
num2 = float(input("Enter the second number: "))
num3 = float(input("Enter the third number: "))
result = calculate_sum(num1, num2, num3)
print(f"The result is: {result}")
```

Output



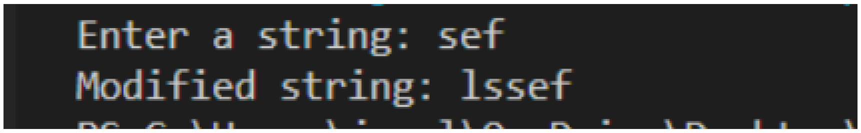
```
Enter the first number: 34
Enter the second number: 67
Enter the third number: 64
The result is: 165.0
```

4.Write a python function to get a new string from a given string where “Is” has been added to the front.If the given string already begins with “Is” then return the string unchanged

Code:

```
def add_ls_to_string(input_string):
if input_string.startswith("ls"):
return input_string
else:
new_string = "ls" + input_string
return new_string
input_str = input("Enter a string: ")
result = add_ls_to_string(input_str)
print("Modified string:", result)
```

Output:



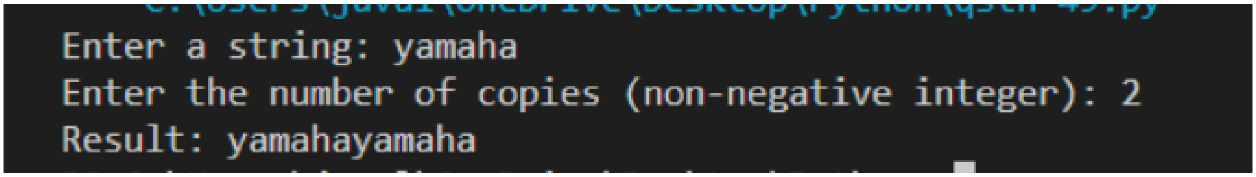
```
Enter a string: sef
Modified string: lssef
```

5.Write a python program to get a string which is n(non-negative integer) copies of a given string

Code:

```
def create_copies(original_string, n):
if n < 0:
return "Invalid input: n should be a non-negative integer."
else:
result_string = original_string * n
return result_string
input_string = input("Enter a string: ")
num_copies = int(input("Enter the number of copies (non-negative integer): "))
result = create_copies(input_string, num_copies)
print("Result:", result)
```

Output:



```
Enter a string: yamaha
Enter the number of copies (non-negative integer): 2
Result: yamahayamaha
```

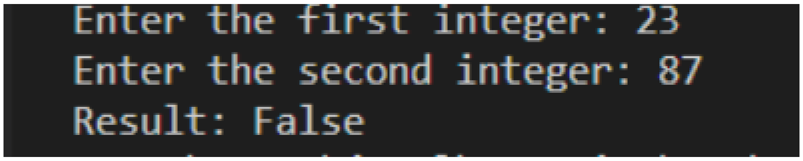
6.Write a python function that will return true if the two given integer values are equal or their sum or difference is 5

Code:

```
def check_values(num1, num2):
if num1 == num2 or num1 + num2 == 5 or abs(num1 - num2) == 5:
return True
```

```
else:
return False
num1 = int(input("Enter the first integer: "))
num2 = int(input("Enter the second integer: "))
result = check_values(num1, num2)
print("Result:", result)
```

Output:



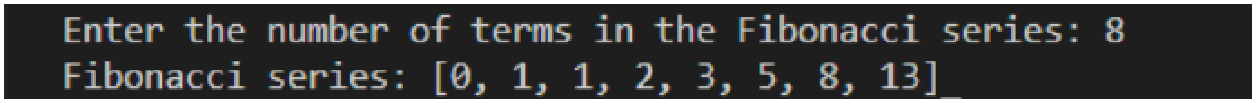
```
Enter the first integer: 23
Enter the second integer: 87
Result: False
```

9.Write a python program to display Fibonacci series using recursion

Code:

```
def fibonacci(n):
if n <= 0:
return []
elif n == 1:
return [0]
elif n == 2:
return [0, 1]
else:
fib_seq = fibonacci(n - 1)
fib_seq.append(fib_seq[-1] + fib_seq[-2])
return fib_seq
n = int(input("Enter the number of terms in the Fibonacci series: "))
result = fibonacci(n)
print("Fibonacci series:", result)
```

Output:



```
Enter the number of terms in the Fibonacci series: 8
Fibonacci series: [0, 1, 1, 2, 3, 5, 8, 13]_
```

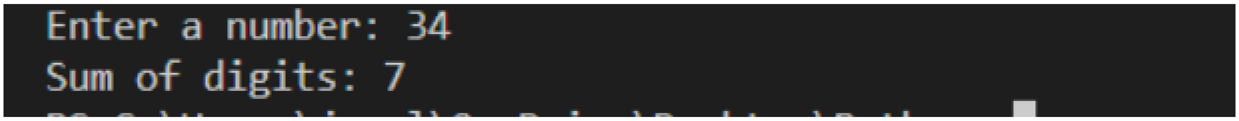
10.Write a python function to find the sum of digits of a number.

Code:

```
def sum_of_digits(number):
digit_sum = 0
while number > 0:
digit = number % 10
digit_sum += digit
number //= 10
return digit_sum
```

```
num = int(input("Enter a number: "))
result = sum_of_digits(num)
print("Sum of digits:", result)
```

Output:



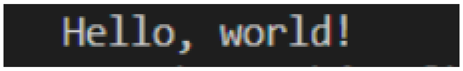
```
Enter a number: 34
Sum of digits: 7
```

11.Write a python function to concatenate two strings.

Code:

```
def concatenate_strings(str1, str2):
concatenated_string = str1 + str2
return concatenated_string
string1 = "Hello, "
string2 = "world!"
result = concatenate_strings(string1, string2)
print(result)
```

Output:



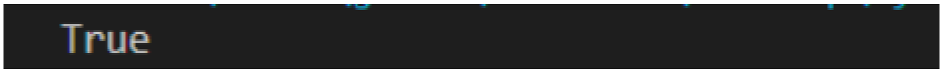
```
Hello, world!
```

12.Write a python function called compare which takes two strings s1 and s2 and an integer n as arguments. The function should return True if the first n characters of both the strings are the same else the function should return False.

Code:

```
def compare(s1, s2, n):
if s1[:n] == s2[:n]:
return True
else:
return False
string1 = "example"
string2 = "examine"
n = 3
result = compare(string1, string2, n)
print(result)
```

Output:



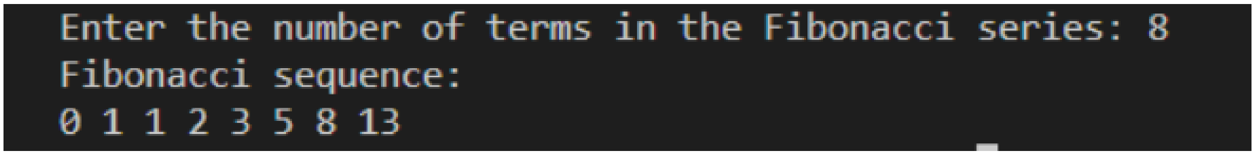
```
True
```

14.Write a python program to display Fibonacci series using recursion

Code:

```
def fibonacci(n):
if n <= 0:
return []
elif n == 1:
return [0]
elif n == 2:
return [0, 1]
else:
fib_seq = fibonacci(n - 1)
fib_seq.append(fib_seq[-1] + fib_seq[-2])
return fib_seq
def display_fibonacci_sequence(n):
fib_sequence = fibonacci(n)
if len(fib_sequence) > 0:
print("Fibonacci sequence:")
for num in fib_sequence:
print(num, end=" ")
print() # Print a newline
else:
print("Invalid input. Please enter a positive integer for the number of terms.")
num_terms = int(input("Enter the number of terms in the Fibonacci series: "))
display_fibonacci_sequence(num_terms)
```

Output



```
Enter the number of terms in the Fibonacci series: 8
Fibonacci sequence:
0 1 1 2 3 5 8 13
```