1.Write a program to Print Fibonacci Series using recursion.

Recursive fibonacci.py - C:/Users/jayan/OneDrive/Documents/DAA/Recursive fibonacci.py (3.12.2)

File Edit Format Run Options Window Help

```
def fibo(n):
    if n<=1:
        return n
    else:
        return (fibo(n-1)+(n-2))
n=10
if n<=0:
    print("Please enter a positive integer:")
else:
    print("Fibonacci sequence: ")
    for i in range(n):
        print(fibo(i))
```

```
Python 3.12.2 (tags/v3.12.2:6abddd9, Feb  6 2024, 21:26:36) [MSC v.1937 64 bit (AMD
n32
Type "help", "copyright", "credits" or "license()" for more information.
>>>
= RESTART: C:/Users/jayan/OneDrive/Documents/DAA/Recursive fibonacci.py
Fibonacci sequence:
0
1
1
2
4
7
11
16
22
29
>>>
```

2. Write a program to check the given no is Armstrong or not using recursive function.

armsrong no.py - C:/Users/jayan/OneDrive/Documents/DAA/armsrong no.py (3.12.2)

File Edit Format Run Options Window Help

```
def arms(num,n1,sum,temp):
    if temp==0:
        if sum==num:
            return True
        else:
            return False
    digit=temp%10
    sum=sum+digit**n1
    temp=temp//10
    return arms(num,n1,sum,temp)
num=int(input("enter no"))
sum=0
n1=len(str(num))
temp=num
res=arms(num,n1,sum,temp)
if res:
    print(num,"armstrong no")
else:
    print(num,"not armstrong no")
```

```
Python 3.12.2 (tags/v3.12.2:6abddd9, Feb  6 2024, 21:26:36) [MSC v.1937
AMD64)] on win32
Type "help", "copyright", "credits" or "license()" for more information.
>>>
= RESTART: C:/Users/jayan/OneDrive/Documents/DAA/armsrong no.py
enter no 153
153 armstrong no
>>>
```

3. Write a program to find the GCD of two numbers using recursive factorization

GCD.py - C:/Users/jayan/OneDrive/Documents/DAA/GCD.py (3.12.2)

File Edit Format Run Options Window Help

```
def gcd(a,b):
    if (b==0):
        return a
    else:
        return gcd(b,a%b)
a=int(input("enter a:"))
b=int(input("enter b:"))
GCD=gcd(a,b)
print("GCD:",GCD)
```

IDLE Shell 3.12.2

File Edit Shell Debug Options Window Help

```
Python 3.12.2 (tags/v3.12.2:6abddd9, Feb  6 2024, 21:26:36) [MSC v.1937 64 bit (AMD64)] on win32
Type "help", "copyright", "credits" or "license()" for more information.
>>>
= RESTART: C:/Users/jayan/OneDrive/Documents/DAA/GCD.py
enter a: 21
enter b: 28
GCD: 7
>>>
```

4. Write a program to get the largest element of an array.

largest.py - C:/Users/jayan/AppData/Local/Programs/Python/Python312/largest.py (3.12.2)

File Edit Format Run Options Window Help

```
def largest(arr, n):
        max = arr[0]

        for i in range(1, n):
                if arr[i] > max:
                        max = arr[i]
        return max
arr = [10, 324, 45, 90, 9808]
n = len(arr)
Ans = largest(arr, n)
print("Largest in given array ", Ans)
```

IDLE Shell 3.12.2

File Edit Shell Debug Options Window Help

```
Python 3.12.2 (tags/v3.12.2:6abddd9, Feb  6 2024, 21:26:36) [MSC v.1937 64 bit (AMD64)
n win32
Type "help", "copyright", "credits" or "license()" for more information.
>>>
== RESTART: C:/Users/jayan/AppData/Local/Programs/Python/Python312/largest.py ==
Largest in given array  9808
>>>
```

5. Write a program to find the Factorial of a number using recursion.

Recursive factorial.py - C:/Users/jayan/OneDrive/Documents/DAA/Recursive factorial.py (3.12.2)

File Edit Format Run Options Window Help

```
def recur_factorial(n):
   if n == 1:
      return n
   else:
      return n*recur_factorial(n-1)

num = 5
if num < 0:
   print("Sorry, factorial does not exist for negative numbers")
elif num == 0:
   print("The factorial of 0 is 1")
else:
   print("The factorial of", num, "is", recur_factorial(num))
```

```
Python 3.12.2 (tags/v3.12.2:6abddd9, Feb  6 2024, 21:26:36) [MSC v.1937 64 bit
n32
Type "help", "copyright", "credits" or "license()" for more information.
>>>
= RESTART: C:/Users/jayan/OneDrive/Documents/DAA/Recursive factorial.py
The factorial of 5 is 120
>>>
```

6. Write a program to copy one string from another string using recursion.

Copy 1 string from another.py - C:/Users/jayan/OneDrive/Documents/DAA/Copy 1 string from another.py (3.12.2)

File Edit Format Run Options Window Help

```
def copy_string(source, destination=""):
  if not source:
    return destination
  return copy_string(source[1:], destination + source[0])
source_string = "Hello, world!"
destination_string = copy_string(source_string)
print(destination_string)
```

IDLE Shell 3.12.2

File Edit Shell Debug Options Window Help

```
Python 3.12.2 (tags/v3.12.2:6abddd9, Feb  6 2024, 21:26:36) [MSC v.1937 64 bit (AMD6
n32
Type "help", "copyright", "credits" or "license()" for more information.
>>>
= RESTART: C:/Users/jayan/OneDrive/Documents/DAA/Copy 1 string from another.py
Hello, world!
>>>
```

7. Write a program to print the reverse of a string using recursion.

Recursive reverse string.py - C:/Users/jayan/OneDrive/Documents/DAA/Recursive reverse string.py (3.12.2)

File Edit Format Run Options Window Help

```
def reverse_string(text):
  if not text:
    return
  reverse_string(text[1:])
  print(text[0], end="")

text = "Hello, world!"
reverse_string(text)
```

IDLE Shell 3.12.2

File Edit Shell Debug Options Window Help

```
Python 3.12.2 (tags/v3.12.2:6abddd9, Feb  6 2024, 21:26:36) [MSC v.1937 64 bit (AMD64)] o
n32
Type "help", "copyright", "credits" or "license()" for more information.
>>>
= RESTART: C:/Users/jayan/OneDrive/Documents/DAA/Recursive reverse string.py
!dlrow ,olleH
>>>
```

8. Write a program to generate all the prime numbers using recursion.

Prime number.py - C:/Users/jayan/OneDrive/Documents/DAA/Prime number.py (3.12.2)

File Edit Format Run Options Window Help

```
def is_prime(num):
  if num <= 1:
    return False
  for i in range(2, int(num**0.5) + 1):
    if num % i == 0:
      return False
  return True

#prints first 10 primes nos
prime_count = 0
num = 2
while prime_count < 10:
  if is_prime(num):
    print(num)
    prime_count += 1
  num += 1
```

IDLE Shell 3.12.2

File Edit Shell Debug Options Window Help

```
Python 3.12.2 (tags/v3.12.2:6abddd9, Feb  6 2024, 21:26:36) [MSC v.1937 64 bit (
n32
Type "help", "copyright", "credits" or "license()" for more information.
>>>
= RESTART: C:/Users/jayan/OneDrive/Documents/DAA/Prime number.py
2
3
5
7
11
13
17
19
23
29
>>>
```

9.Write a program to check a number is a prime number or not using recursion.

primr.py - C:/Users/jayan/OneDrive/Documents/DAA/primr.py (3.12.2)

File Edit Format Run Options Window Help

```
def Prime_Number(n, i=2):
    if n == i:
        return True
    elif n % i == 0:
        return False
    return Prime_Number(n, i + 1)


n = 971
if Prime_Number(n):
    print("Yes,", n, "is Prime")
else:
    print("No,", n, "is not a Prime")
```

IDLE Shell 3.12.2

File Edit Shell Debug Options Window Help

```
Python 3.12.2 (tags/v3.12.2:6abddd9, Feb  6 2024, 21:26:36) [MSC v.1937 64 bit (A
MD64)] on win32
Type "help", "copyright", "credits" or "license()" for more information.
>>>
= RESTART: C:/Users/jayan/OneDrive/Documents/DAA/primr.py
Yes, 971 is Prime
>>>
```

10.Write a program for to check whether a given String is Palindrome or not using recursion

palindrome.py - C:/Users/jayan/OneDrive/Documents/DAA/palindrome.py (3.12.2)

File Edit Format Run Options Window Help

```
def isPalindrome(s):
        return s == s[::-1]
s = "malayalam"
ans = isPalindrome(s)

if ans:
        print("Yes")
else:
        print("No")
```

IDLE Shell 3.12.2

File Edit Shell Debug Options Window Help

```
Python 3.12.2 (tags/v3.12.2:6abddd9, Feb  6 2024, 21:26:36) [MSC v.1937 64 bit (AMD64)] on wi
n32
Type "help", "copyright", "credits" or "license()" for more information.
>>>
= RESTART: C:/Users/jayan/OneDrive/Documents/DAA/palindrome.py
Yes
>>>
```