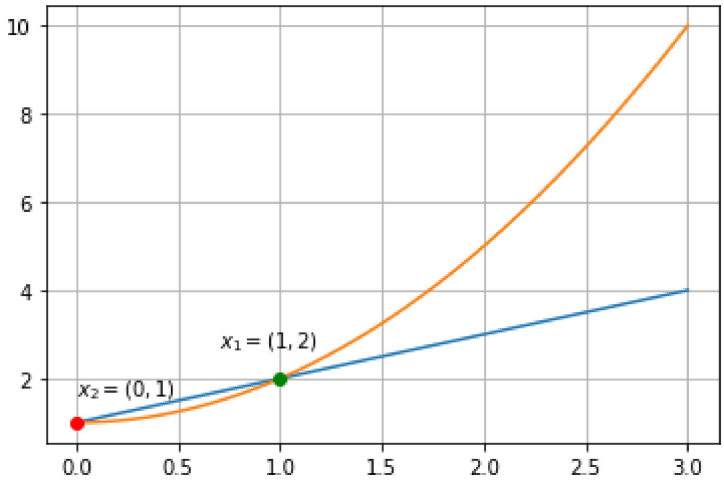
In [5]:
```
import matplotlib.pyplot as plt
import numpy as np

x = np.linspace(0,3,1000)
plt.plot(x,x+1,label = "$x - y = -1$")
plt.plot(x,(x**2)+1,label = "$y = x^{2} + 1$")
plt.grid()


plt.plot(1,2,'go')

plt.annotate('$x_{1} = (1,2)$',(0.7,2.7))
plt.plot(4.03896783e-28,1,'ro')
plt.annotate('$x_{2} = (0,1)$',(0,1.6))

plt.show()
```



In [ ]:

1) $y = 2x + 2; y = x - 1$ 2) $y = -2x - 7; y = x + 9$ 3) $y = -1; (x - 2)^2 + (y + 3)^2 = 4$ 4) $x = -6; (x + 3)^2 + (y - 1)^2 = 9$

In [ ]:

to find sum of series

1) 1 + 2 + 3 + ... n

In [2]:
```
n = int(input("Enter the range of number: "))


sum = 0
for i in range(1, n+1):
    sum+=i

print("The sum of the series = ",sum)
```

```
The sum of the series =  36
```

In [ ]:

2) 1 - 2 + 3 - ... n

In [4]:
```
n = int(input("Enter the range of number: "))

print("n = ",n)
sum = 0
for i in range(1, n+1):
    if i%2 == 0:
        sum-=i
    else:
        sum+=i

print("The sum of the series = ",sum)
```

```
The sum of the series =  3
```

3) 1^3 + 2^3 + 3^3 ... n

In [5]:
```
n = int(input("Enter the range of number: "))

print("n = ",n)
sum = 0
for i in range(1, n+1):
    sum+=i**3

print("The sum of the series = ",sum)
```

```
The sum of the series =  36
```

Type *Markdown* and LaTeX: α^2

4) 1 + 2 - 3 - 4 + 5 + 6... n

In [ ]:

In [12]:
```
n = int(input("Enter the range of number: "))

print("n = ",n)
sum = 0
c= 1
for i in range(1, n+1,2):
    if c%2 == 0:
        sum -= i
        if i+1<=n:

            sum -= i+1
    else:
        sum+=i
        if i+1<=n:
            sum+= i+1
    c+=1


print("The sum of the series = ",sum)
```

```
n =  4
The sum of the series =  -4
```

5) 2 + 4 + 8 ... n

In [13]:
```
n = int(input("Enter the range of number: "))
print("n = ", n)

sum = 0
for i in range(1, n+1):
    sum+= 2**i

print("The sum of the series = ",sum)
```

```
n =  3
The sum of the series =  14
```

Series

1. $\sum_{n=1}^{\infty} \frac{x^n}{n}$

In [14]:
```
def f(x,i):
    return x**i/i

n = int(input("Enter number of terms : "))
print("n = ", n)

x = int(input("Enter value of x: "))
print("x = ",x)

s = 0

for i in range(1,n+1):
    s = s + f(x,i)

print("sum of series is  = ",s)
```

```
n =  5
x =  1
sum of series is  =  2.283333333333333
```

2. $\sum_{n=1}^{\infty} (-1)^{n+1} \frac{x^n}{n}$

In [15]:
```
def f(x,i):
    return ((-1)**(i+1))*(x**i/i )

n = int(input("Enter number of terms : "))
print("n = ", n)

x = int(input("Enter value of x: "))
print("x = ",x)

s = 0

for i in range(1,n+1):
    s = s + f(x,i)

print("sum of series is  = ",s)
```

```
n =  2
x =  1
sum of series is  =  0.5
```

In [ ]:
```

```

3. $\sum_{n=1}^{\infty} \frac{1}{n^2-1}$

In [3]:
```
sum([1/(i**2 -  1) for i in range(2,10)])
```

Out[3]:
```
0.6444444444444444
```

printing the series x + x^2/2 + x^3/3 ....

In [6]:
```
import sympy as sp
from sympy import pprint
x = sp.Symbol('x')
series = x

n = int(input("Enter the number of terms you want in the series: "))
print(n)
for i in range(2,n+1):
    series = series + (x**i)/i

pprint(series)
series
```

```
5
 5    4    3    2
x    x    x    x
── + ── + ── + ── + x
5    4    3    2
```

Out[6]: $\frac{x^5}{5} + \frac{x^4}{4} + \frac{x^3}{3} + \frac{x^2}{2} + x$

for the right order

In [8]:
```
from sympy import Symbol, pprint, init_printing

def print_series(n):
    # initializing reverse order
    init_printing(order = 'rev-lex')
    x = Symbol('x')
    series = x
    for i in range(2,n+1):
        series = series + (x**i)/i
    return series

n = int(input('Enter the number of terms : '))
print(n)
print_series(n)
```

```
5
```

Out[8]: $x + \frac{x^2}{2} + \frac{x^3}{3} + \frac{x^4}{4} + \frac{x^5}{5}$

Calculating the value of a series

in addition to printing the series, we want our program to be able to find the value of the series for a particular value

Type *Markdown* and LaTeX: α^2

In [26]:
```
from sympy import Symbol, pprint, init_printing

def print_series(n,x_value):
    # initializing reverse order
    init_printing(order = 'rev-lex')
    x = Symbol('x')
    series = x
    for i in range(2,n+1):
        series = series + (x**i)/i
    series_value = series.subs({x:x_value})
    print('value of series at x value at', x_value,' = ', series_value)
    return series

n = int(input('Enter the number of terms : '))
print(n)
x = int(input('enter value of x : '))
print('x = ',x)
print_series(n,x)
```

```
3
x =  1
value of series at x value at 1  =  11/6
```

Out[26]:

$$x + \frac{x^2}{2} + \frac{x^3}{3}$$

fibonacci series

In [23]:
```
def fib(n, memo={}):
    if n in memo:
        return memo[n]
    if n == 0:
        return 0
    if n<=2:
        return 1
    memo[n] = fib(n - 1, memo) + fib(n - 2, memo)
    return memo[n]

n = int(input('enter the value of n : '))

p = 0

while fib(p)<=n:
    print(fib(p),end = " ")
    p+=1
```

```
0 1 1 2 3 5 8 13
```

9 + 99 + 999 .....

In [30]:
```
n = int(input("enter the value of n : "))
print('n = ', n)
s = 0
su = 0
p = 0
while p<n:
    s = s*10 + 9
    su+=s
    p+=1

su
```

```
n =  2
```

Out[30]: 108

1! + 2! + 3! ...

In [33]:
```
n = int(input('enter the value of n : '))
print(n)
s = 0
fact = 1
for i in range(1,n+1):
    fact*=i
    s+=fact
print("sum = ", s)
```

```
3
sum =  9
```

1 + (1 +3) + (1 + 3 + 5)...

In [42]:
```
n =  int(input('enter the range: '))
p = 0
s = 0
su = 0
odd = 1

while p<n:
    s+=odd
    odd+=2

    su+=s
    print(s,end = " + ")
    p+=1
print()
print(su)
```

```
1 + 4 + 9 + 16 +
30
```

In [43]:
```
999 + 108
```

Out[43]: 1107