Experiment:4

INTRODUCTION TO SYMBOLIC COMPUTING

(NumPy and SimPy)

1. Write a function in Python that generates a (10x10) matrices with entries as $a_{ij} = (i + j)^2$ with i and j are random integers from 0 to 4 (where the user provides these entries as input). Additionally, within the same function, select a square sub matrix of order (5×5) from the generated matrix of random order, and then compute the result of the square of that sub matrix.

Solution:

```
random_start=int(input())
random_end=int(input())
order=int(input())
def rand_matrix(random_start,random_end,order):
    A=np.zeros((order,order))
    for i in range(0,9):
        for j in range(0,9):
            n3=np.random.randint(random_start,random_end)
            n4=np.random.randint(random_start,random_end)
            A[i,j]=np.round(((n3+n4)**2)/2,decimals=4)
    n5=np.random.randint(1,5)
    n6=np.random.randint(1,5)
    sub_mat=A[n5:n5+5,n6:n6+5]
    matrix_mul=np.round(np.matmul(sub_mat,sub_mat),decimals=4)
    print(matrix_mul)
rand_matrix(random_start,random_end,order)
```

2. Define the following function in the variables x and y
   1. $f(x) = 2sin^2(x) + log(x)$ and evaluate f when $x = 10$ and $x = \pi$
   2. Find the derivative of $sin(x) + x^3 + 2x + cos(4x)$
   3. $g(x, y) = \dfrac{x^2}{y} + \dfrac{y^3}{x + y}$ and evaluate g when $x = -2$ and $y = -7$.

Solution:

```
 import sympy as sp
from sympy import I,pi,oo
x=sp.Symbol("x")
y=sp.Symbol("y")
#1
f=2*(sp.sin(x))**2+sp.log(x)
print(f)
values1={x:10}
print(sp.N(f.subs(values1)))
values2={x:pi}
print(sp.N(f.subs(values2)))
#2
expr=sp.sin(x)+x**3+2*x+sp.cos(4*x)
expr
der=expr.diff(x)
#3
g=x**2/y+y**3/(x+y)
g
values3={x:-2,y:-7}
print(sp.N(g.subs(values3)))
```