PROGRAM 3

AIM : To write a python program to find the integral of a polynomial.

ALGORITHM :

- **Input Degree**: The program will first take the degree of the polynomial. If the degree is zero or negative, it outputs "Invalid Input".
- **Input Coefficients**: The program will prompt the user for the coefficients of the polynomial starting from the highest degree.
- **Integration**: The program integrates each term of the polynomial:
  For a term a*x^n, the integral is (a/(n+1)) * x^(n+1).
  If a is divisible by n+1, the output should be a simpler form.
  Otherwise, the integral should include a fraction.
- **Output the Integrated Polynomial**: It outputs the integrated polynomial, properly formatted, and includes + C at the end to indicate the constant of integration.

PROGRAM CODE :

```
deg=int(input("Enter the degree of polynomial: "))
if (deg<=0):
  print("Invalid Input")
else:
  brr=[]
  for i in range(deg+1):
    print("Enter the coefficient of x^",deg-i)
    coeff=int(input())
    brr.append(coeff)

  print("integrated equation is :")
  for i in range(deg+1):
    exp=deg-i+1
    coe=brr[i]
    if (exp==1):
      print(" (",coe,"x) + C")
    elif (coe%exp==0):
      print(" (",int(coe/exp),"x^",exp,") + ",end='')
    else:
      print(" (",coe,"x^",exp,"/",exp,") + ",end='')
```

REFUTE TEST CASE :

- **Entered degree of the polynomial is not an integer : Data type will not be matched as the input it taken in integer type so an appropriate error will be displayed.**
- **Entered degree of the polynomial is zero or a negative value : Invalid Input will be displayed as degree cannot be less than one.**

TEST RUN 1 :

```
>>>
= RESTART: C:\Users\Smile\AppData\Local\Programs\Python\Python312\new.py
Enter the degree of polynomial: 4.0
Traceback (most recent call last):
  File "C:\Users\Smile\AppData\Local\Programs\Python\Python312\new.py", line 1, in <module>
    deg=int(input("Enter the degree of polynomial: "))
ValueError: invalid literal for int() with base 10: '4.0'
>>>
```

TEST RUN 2 :

```
>>>
= RESTART: C:\Users\Smile\AppData\Local\Programs\Python\Python312\new.py
Enter the degree of polynomial: -3
Invalid Input
>>>
```

TEST RUN 3 :

```
>>>
= RESTART: C:\Users\Smile\AppData\Local\Programs\Python\Python312\new.py
Enter the degree of polynomial: 3
Enter the coefficient of x^ 3
4
Enter the coefficient of x^ 2
5
Enter the coefficient of x^ 1
6
Enter the coefficient of x^ 0
7
integrated equation is :
 ( 1 x^ 4 ) +  ( 5 x^ 3 / 3 ) +  ( 3 x^ 2 ) +  ( 7 x) + C
>>>
```

RESULT : The program code was run and executed successfully.

PROGRAM 4:

AIM : To write a python program to find Eigen values and Eigen vectors of a given matrix.

ALGORIHTM :

- **Input: Number of rows and columns from the user.**
- **Validation: Check if inputs are valid (non-negative and non-zero).**
- **Matrix Creation: Collect matrix values row by row.**
- **Output: Print the original matrix.**
- **Eigen Calculation: Use NumPy to compute eigenvalues and eigenvectors.**
- **Display Output: Print eigenvalues and eigenvectors.**

PROGRAM CODE :

```
import numpy as np
row=int(input("Enter  number of rows: "))
col=int(input("Enter the number of coloumns : "))
if (row<0 or col<0):
   print("Invalid input")
elif (row==0 or col==0):
   print("Entered value is zero thus no further operation will be done")
else:
   matrix=[]
   for i in range(row):
      a=[]
      for j in range(col):
         print("Enter the value of(",i,",",j,") : ")
         x=int(input())
         a.append(x)
      matrix.append(a)
   print("Printing the Original square array:\n")
   for i in matrix:
      print(i)
   w, v = np.linalg.eig(matrix)
   print("Printing the Eigen values of the given matrix:\n",w)
   print("Printing Right eigenvectors of the given matrix:\n",v)
```

REFUTE TEST CASE :

- Entered number of rows or coloumns is zero : A message is displayed that no further operation will be done because no value will be entered in the matrix.
- Entered number of rows or columns is negative : Invalid input will be displayed.
- Entered number of rows or column is not an integer : Data type will not be matched and compiler will give an error.

TEST RUN 1 :

```
>>>
= RESTART: C:\Users\Smile\AppData\Local\Programs\Python\Python312\new.py
Enter  number of rows: 0
Enter the number of coloumns : 0
Entered value is zero thus no further operation will be done
>>>
```

TEST RUN 2 :

```
>>>
= RESTART: C:\Users\Smile\AppData\Local\Programs\Python\Python312\new.py
Enter  number of rows: 2
Enter the number of coloumns : 2
Enter the value of( 0 , 0 ) :
1
Enter the value of( 0 , 1 ) :
2
Enter the value of( 1 , 0 ) :
3
Enter the value of( 1 , 1 ) :
4
Printing the Original square array:

[1, 2]
[3, 4]
Printing the Eigen values of the given matrix:
 [-0.37228132  5.37228132]
Printing Right eigenvectors of the given matrix:
 [[-0.82456484 -0.41597356]
 [ 0.56576746 -0.90937671]]
>>>
```

TEST RUN 3 :

```
>>>
= RESTART: C:\Users\Smile\AppData\Local\Programs\Python\Python312\new.py
Enter  number of rows: 4.0
Traceback (most recent call last):
  File "C:\Users\Smile\AppData\Local\Programs\Python\Python312\new.py", line 2, in <module>
    row=int(input("Enter  number of rows: "))
ValueError: invalid literal for int() with base 10: '4.0'
>>>
```

TEST RUN 4:

```
>>>
= RESTART: C:\Users\Smile\AppData\Local\Programs\Python\Python312\new.py
Enter  number of rows: -3
Enter the number of coloumns : 2
Invalid input
>>>
```

RESULT : The program was run and executed successfully.