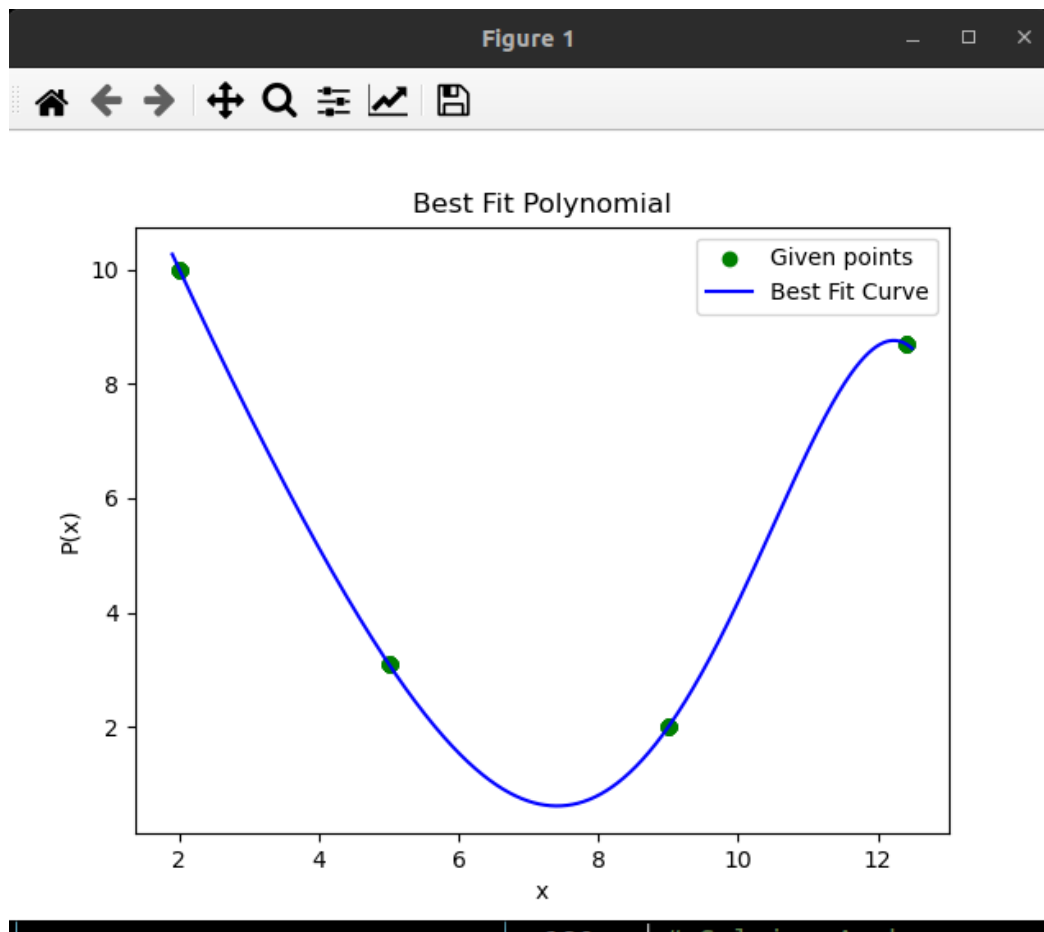112001041
Sidharth chadha

Lab report week-5

1)
We run a loop and for each iteration compute a polynomial with their coefficients using the tuples provided in the input.
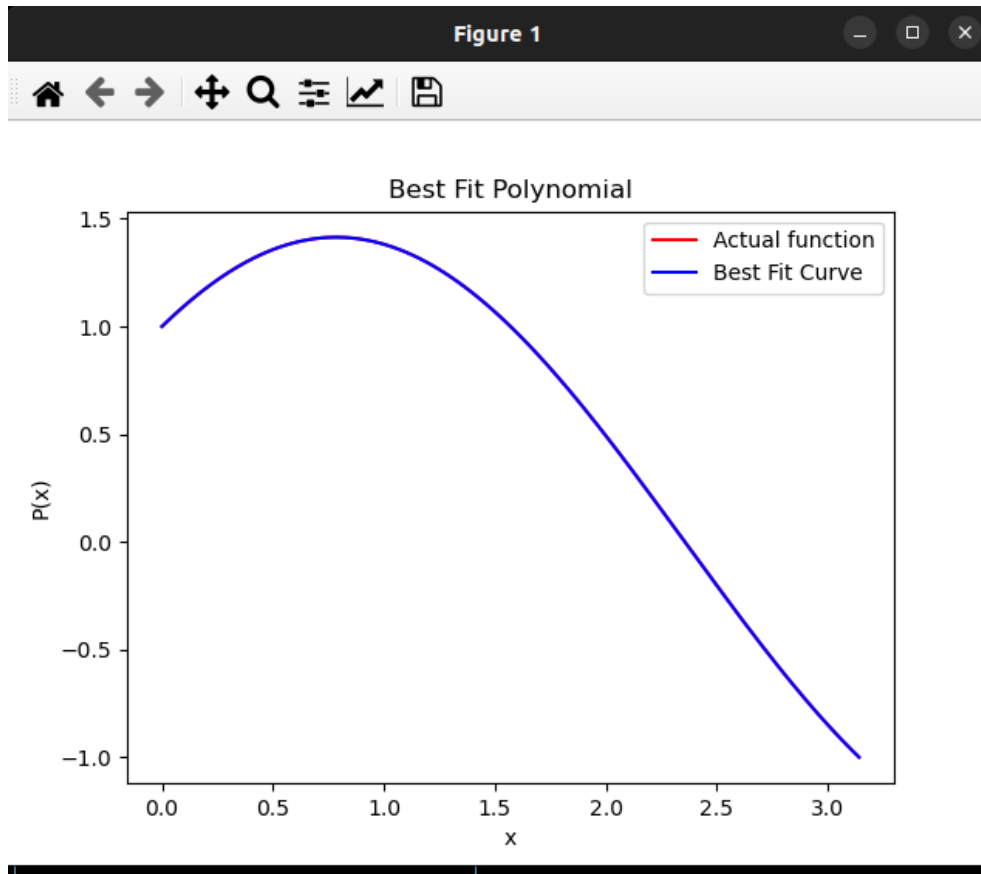
Example input and output
```
((9,2),(2,10),(12.4,8.7),(5, 3.10))
```


2)
This is similar to 1st question just replacing summing with integration. We use approximate integral (trapezoidal area) to calculate the coeffs.

Best Fit Polynomial

3)

We are using `Ln(x) = (1/(2^n*n!))*(dn/dx^n(x^2 - 1)^n)`
We are using derivative created in previous labs (for polynomial)
to differentiate in n times.

```
189
190    l = Legendre(15)
191    print(l)
192
```

PROBLEMS   OUTPUT   DEBUG CONSOLE   TERMINAL   COMMENTS   bash - week5

● (base) sid@sid-HP-Spectre-x360-Convertible-13-aw0xxx:~/Documents/cma/week5$ python3 q3.py
0.0 -3.14208984375 0.0 124.63623046875 0.0 -1420.85302734375 0.0 7104.265136718749 0.0 -18155.34423828125 0.0 24757.28759765625 0.0 -17139.6606
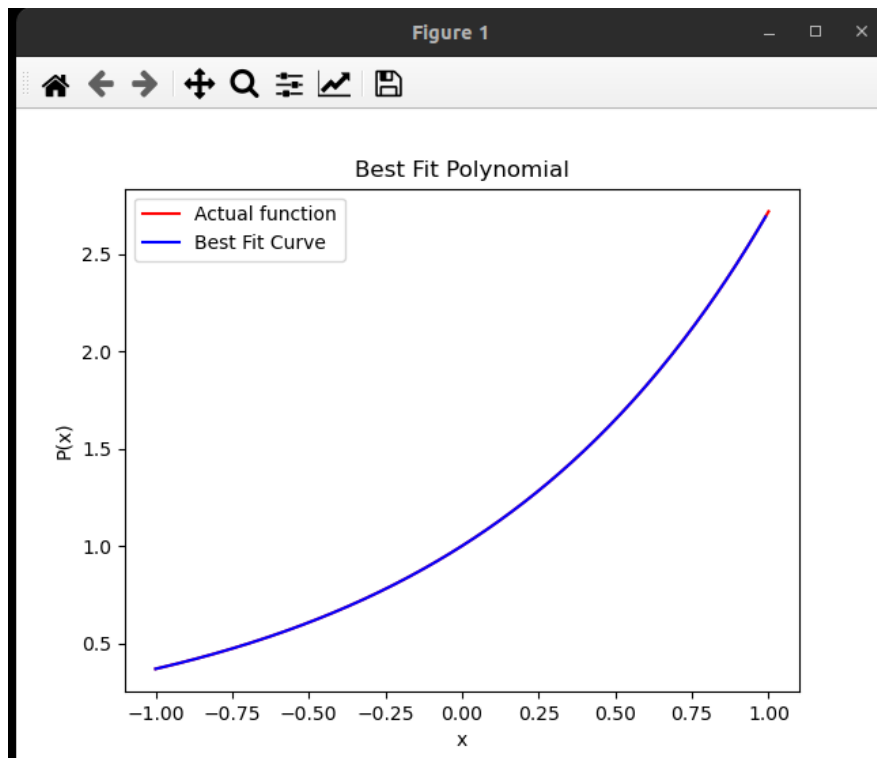4453125 0.0 4733.81103515625

4)

We will use the n legendre polynomials with the below formula

$$Q_n(x) = \sum_{i=0}^{n} a_i \phi_i(x)$$

We then plot the approximation and actual function in range [-1,1]

Example:

```
leastSquare(7)
```

Best Fit Polynomial

5)
Function chebyshev generates nth chebyshev polynomial recursive relation


```
155
156    t = Chebyshev(15)
157    print(t)
```

PROBLEMS   OUTPUT   DEBUG CONSOLE   TERMINAL   COMMENTS

KeyboardInterrupt

(base) sid@sid-HP-Spectre-x360-Convertible-13-aw0xxx:~/Documents/cma/week5$ python3 q5.p
0 -15 0 560 0 -6048 0 28800 0 -70400 0 92160 0 -61440 0 16384
(base) sid@sid-HP-Spectre-x360-Convertible-13-aw0xxx:~/Documents/cma/week5$ ▌

active issue   Sign in to Bitbucket   0 0   Live Share   Ln 13

6)

We calculate the product of 2 chebychev polynomials at a given value anf then calculate the approximate integral of their product.
For every pair of such polynomials we calculate this integral and find out the answer.

7)the function calculate_coeffiecients calculate the fourrier coefficients for the given n and calculate_fourier uses these to calculate the fourier transformation.
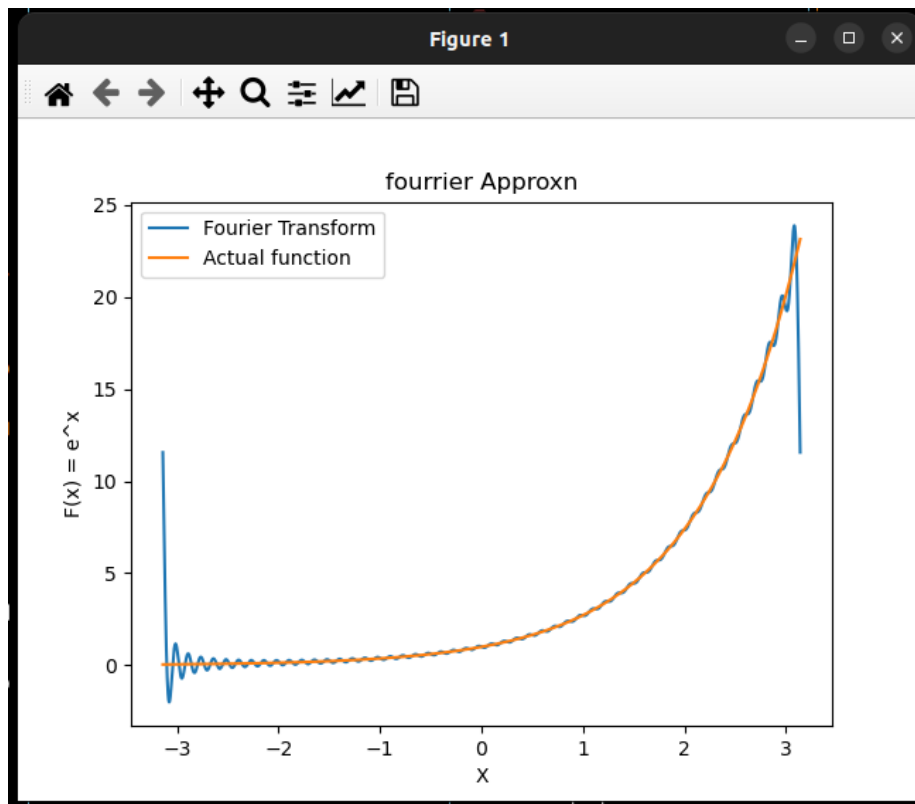
8)
fft and ifft takes O(nlogn).
we convert two numbers to their fourier polynomial. Now we multiplied the corresponding values of the arrays to get the respective product.