Exercise 1

Write a program that can calculate $(a_1x^3 + b_1x^2 + c_1x + d_1)/(a_2x^3 + b_2x^2 + c_2x + d_2)$. The inputs of your program are the coefficients of dividend and divisor. Then the outputs are the coefficients of quotient and remainder.

Requirements

a) The coefficients of each polynomials are separated by **Space** and ended by **Enter**. The first number is coefficient of the highest-order component and the last one is a constant. The inputs can be represented as

- Dividend : a_1 b_1 c_1 d_1
- Divisor : a_2 b_2 c_2 d_2

b) The inputs are all integers, and a_1 should be nonzero. It is required to report the errors for illegal inputs. Then the outputs are integers or fractions of simplified form. For example, if we want to calculate $(2x^3 + 4x^2 + 2x)/(4x + 4)$, the inputs and outputs are given by

- Dividend : 2 4 2 0
- Divisor : 0 0 4 4
- Quotient : 1/2 1/2 0
- Remainder: 0

Test Points

a) Test Input:
1 -7 10 3
0 0 1 -5
Expected Output:
1 -2 0
3

b) Test Input:
4 -4 1 3
0 0 3 0
Expected Output:
4/3 -4/3 1/3
3

c) Test Input:
3 -2 3/5 0
0 1 7 5
Expected Output:
error

Exercise 2

Below we will give several distinct numbers $x_0, x_1, \ldots, x_n$ and the corresponding values of the function $f(x_0), f(x_1), \ldots, f(x_n)$. Use the Newton interpolation to approximate other values. This part is covered by the *NumericalAnalysis* [1] from page 124 to 129.

Requirements

Define a class *NewtonInterpolation* that satisfies the following operations:

a) Store Table 1 as its member variable.

b) Define a member function $DividedDifferenceTable$ to compute the divided difference table and print it to the screen. (**Note**: We will not limit the output format but it should be in a tabular form in order to be recognized easily, printing each entry in one line is not allowed.)

c) Define a member function $ApproximateValue$ to obtain an approximate value at new point x_{new}. **Hint:** The coefficients of the polynomial have been computed above.

Table 1

x	f(x)
0.1	4.2
0.4	3.8
1.0	2.5
1.8	2.0
2.9	-2.0
3.2	0.0
3.8	-0.3

After finishing *NewtonInterpolation* class, test it in your main function. First, calling the $DividedDifferenceTable$ to print the divided difference table. Then waiting for the user input and calling $ApproximateValue$ to get the approximate results. Quit the program if there is any non-numerical input.

Test Points

- Divided difference table such as

```
Jiahuis-MacBook-Pro:assignment
1
2 1
4 2 0.5
8 4 1 0.1667
```

- Test Input: 1.6
  Test Output: 2.55439
- Test Input: 2.8
  Test Output: -2.26106
- Test Input: 0.7
  Test Output: 2.77756

References

[1] R.L.Burden, J.D.Faires, Numerical Analysis, 9th Edition,
https://fac.ksu.edu.sa/sites/default/files/numerical$_a nalysis_9 th.pdf$