Problem A: Polynomial

Description

In mathematics, a polynomial is an expression of finite length constructed from variables and constants, using only the operations of addition, subtraction, multiplication, and non-negative integer exponents. For example, $4x^2 - x + 5$ is a polynomial, but $x^2 + \frac{4}{x} + \frac{3}{2}$ is not, because its second term involves division by the variable x ($\frac{4}{x}$) and because its third term contains an exponent that is not an integer.

In this problem, your job is to implement the polynomial addition, polynomial multiplication, polynomial function derivation, polynomial function definite integral in PolySeq class.

Specification

You must implement the PolySeq class with the following public member functions:

Functions	**Description**
PolySeq(int *,int *)	Constructor. The parameters mean coefficient and exponent respectively.
PolySeq( )	Constructor with no parameter.
PolySeq Add(PolySeq)	Return the sum of two polynomials.
PolySeq Multiply(PolySeq)	Return the product of two polynomials.
PolySeq Derivative( )	Return the derivative of the polynomial function.
int Integral(int ,int);	Return the result of the definite integral of the polynomial function. The parameters mean the lower bound and the upper bound of the integral respectively.
int getvalue(int);	Return the result of the polynomial with the specified parameter.

For example: P1 = $6x+1$, P2= $3x^2+3x+2$

Functions	**Mathematical expression**
P1. Add(P2)	$(6x+1)+(3x^2+3x+2)$
P1. Multiply(P2)	$(6x+1)*(3x^2+3x+2)$
P1. Derivative( )	$(6x+1)'$
P1.Integral(2 ,3);	$\int_2^3 6x+1$

Sample Main function

```
int main()
{
    int n1,n2,x1,x2;
    cin >> n1;
    int *c1=new int[n1];
    int *e1=new int[n1];
    for(int i=0;i<n1;i++){
        cin >> c1[i];
    }
    for(int i=0;i<n1;i++){
        cin >> e1[i];
    }
    cin >> n2;
    int *c2=new int[n2];
    int *e2=new int[n2];
    for(int i=0;i<n2;i++){
        cin >> c2[i];
    }
    for(int i=0;i<n2;i++){
        cin >> e2[i];
    }
    cin >> x1;
    cin >> x2;
```

```
    PolySeq P1(c1,e1,n1);
    PolySeq P2(c2,e2,n2);

    PolySeq P3=P1.Add(P2);
    cout << P3.getvalue(x1)
         << endl;

    PolySeq P4=P1.Multiply(P2);
    cout << P4.getvalue(x1)
         << endl;

    PolySeq P5=P1.Derivative();
    cout << P5.getvalue(x1)
         << endl;

    cout << P2.Integral(2,3);

    delete [ ]c1;
    delete [ ]e1;
    delete [ ]c2;
    delete [ ]e2;

    return 0;
}
```

Input

The first line contains an integer n1 indicating the number of terms of the first polynomial. The second line and third line contain the elements of arrays c1[ ], e1[ ], where c1[i] and e1[i] (0<=i<n1) are coefficient and exponent of the first polynomial respectively. Both c1[i] and e1[i] are in the range of integer value and each number is separated by a space.

The forth line contains an integer n2 indicating the number of terms of the second polynomial. The fifth line and sixth line contain the elements of arrays c2[ ], e2[ ], where c2[i] and e2[i] (0<=i<n2) are coefficient and exponent of the second polynomial respectively. Both c2[i] and e2[i] are in the range of integer value and each number is separated by a space.

The seventh line contains two integers x1 and x2.

Output

The output should print the following integers in order.

(1) The sum of the first and the second polynomials with parameter x1.

(2) The product of the first and the second polynomials with parameter x1.

(3) The derivative of the first polynomial with parameter x1.

(4) The result of the definite integral of the second polynomial with parameter lower bound x1 and upper bound x2.

Sample Input	**Sample Output**
4 3 -2 1 0 3 2 1 0 3 9 -4 1 2 1 0 2 3	47 522 29 48

Restriction

1. The code you submitted should only contain the PolySeq class (with no header file and function).
2. In the question, the <iostream> is the only header file allowed.