

SE-1105 HOMEWORK-2

You can represent polynomials with two arrays of the same length as follows:

Let the polynomial be: $P(x) = c_0x^{d_0} + c_1x^{d_1} + c_2x^{d_2} + \dots + c_nx^{d_n}$ where c_i 's are the real-valued coefficients of the terms and d_i 's are increasing sequence of integers. Then the arrays store

`coefficients[] = { $c_0, c_1, c_2, \dots, c_n$ } and`

`degrees[] = { $d_0, d_1, d_2, \dots, d_n$ } (in increasing order)`

Assume that there will be at most 20 terms in the polynomial, and thus, the sizes of the arrays are 20. If the polynomial has fewer terms than 20, the remaining elements in the arrays remain 0. **The polynomials do not store the terms with a coefficient value of 0.0.** So, you can understand the number of terms in the polynomial by looking at the number of non-zero elements at the beginning of the coefficients array.

Example:

If the polynomial : $P(x) = 4x^3 - 3x^2 + 1$ then

`coefficients[] = { 1 -3, 4, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0 } and`
`degrees[] = { 0, 2, 3, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0 }`

If the polynomial : $P(x) = x^7 - x^4 + 2x^3$ then

`coefficients[] = { 2 -1, 1, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0 } and`
`degrees[] = { 3, 4, 7, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0 }`

You are supposed to write a program that includes at least the following functions:
(Assume that the polynomials will never have more than 20 terms)

- `readPolynomial(double coefficients[], int degrees[]):`

Reads the coefficients and degrees of the polynomial from the user by reading one term (one degree and one coefficient) at a time until the user enters the value 0.0 as the coefficient. Store the terms in coefficients and degrees arrays in sorted order according to the degrees of the terms.

- `printPolynomial(double coefficients[] int degrees[]):`

Prints out the given polynomial as we write in mathematics(see examples above). The rules are basically:

- '+' sign is not printed before negative terms.
- You can use '^' sign to denote power. i.e. $3x^2$ means $3x^2$
- Degree 1 is not printed
- For zero-degree terms, x is not printed

- Coefficient 1.0 is not printed
- Coefficient -1.0 is printed as “-”
- `addPolynomials(double c1[], int d1[], double c2[], int d2[], double c3[], int d3[]):`

Adds polynomials represented by c1,d1, and c2,d2 arrays, and stores the result in c3,d3 arrays.

Note that in the resulting polynomial (c3,d3), the terms must be stored in increasing order according to the degrees of the terms, and there should be no term with the same degree.

Also, note that the polynomial should not store the degree of any term with a coefficient value of 0.0.

Example:

If c1,d1 and c2,d2 arrays represent $P(x) = -x^5 + 4x^3 - 3x^2 + 1$ and $Q(x) = x^7 + 2x^5 + 3x^2 - 2x$ as follows

```
c1 []    = {1, -3, 4, -1, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0} and
d1 []    = {0, 2, 3, 5, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0},
c2 []    = {-2, 3, 2, 1, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0} and
d2 []    = { 1, 2, 5, 7, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0}.
```

Then c3,d3 represents $R(x) = x^7 + x^5 + 4x^3 - 2x + 1$ as follows

```
c3 []    = {1, -2, 4, 1, 1, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0} and
d3 []    = {0, 1, 3, 5, 7, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0}
```

- `main():` Reads two polynomials from the user, prints them, and prints their sum.