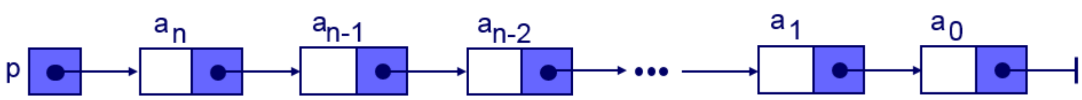
Polynomial Addition

This first practical homework task shall repeat and practice dynamic programming with structures, pointers and lists to prepare later on introduction of classes, constructors, operators, ...
You shall not anticipate programming a class but perform input and output by **`cout`** and **`cin`** (instead of **`printf`** and **`scanf`** in C) as well as use **`new`** and **`delete`** from C++ (instead of **`malloc`** and **`free`** in C).

Polynom $p(x)=a_n x^n+a_{n-1}x^{n-1}+\ldots+a_1x^1+a_0$, $n\geq 0$, $a_n\neq 0$ A polynomial in the field of real numbers is uniquely determined by its coefficients like shown above.

Such a polynomial of an arbitrary degree can be implemented inside a computer by a list of its polynomial coefficients, whereas the first polynomial coefficient a_n of the monomial with highest degree x^n gets first list element and the polynomial coefficient a_0 of the monomial with smallest degree x^0 last list element.



Subtasks

- Define a structure for the polynomial coefficients as list elements and a function with a floating-point number as single parameter to initialize a newly generated list element on the heap and return a pointer to it.
- Write a function with a polynom as parameter returning the polynom's degree (i.e., counts the coefficients/list elements).
- Write a C++ function without parameters to input a polynomial with coefficients a_n, a_{n-1}, ... from standard character input stream one by one, generating a list of above structure and returning the inputted polynomial .
- $p(x)=x^5+2.3x^4-x^2+2.7x-3.8$: **`1*x^5+2.3*x^4-1*x^2+2.7*x^1-3.8`**.,
  Write a C++ function with a pointer to a polynomial (i.e. a list of polynomial coefficients) as parameter to output a polynomial in textual form onto the standard character output stream.
  As example, for a polynomial $p(x)=x^5+2.3x^4-x^2+2.7x-3.8$ the output shall be: **`1*x^5+2.3*x^4-1*x^2+2.7*x^1-3.8`**, further examples see below.
- $p(x)$
  $=a_n\cdot x^n+a_{n-1}\cdot x^{n-1}+\ldots+a_1\cdot x^1+a_0$

$=((\ldots((a_n \cdot x + a_{n-1}) \cdot x + a_{n-2}) \cdot x + \ldots) \cdot x + a_1) \cdot x + a_0.$

Write a C++ function with a pointer to a polynomial p and a point x as parameters calculating the polynomial value $y=p(x)$ by the [Horner method] using only simple multiplications and additions (no predefined functions like **`pow`** shall/need to be used):

y

$=p(x)$

$=a_n \cdot x^n + a_{n-1} \cdot x^{n-1} + \ldots + a_1 \cdot x^1 + a_0$

$=((\ldots((a_n \cdot x + a_{n-1}) \cdot x + a_{n-2}) \cdot x + \ldots) \cdot x + a_1) \cdot x + a_0.$

(Hint: think about, that the given list structure perfectly fits to the Horner method calculation).

- Write a C++ function outputting a table of values of a polynomial in an interval [**`a`**, **`b`**] with step width **`h`** onto the standard character output stream. A polynomial, the two interval borders as well as the step width shall be parameters of the function (output examples see below).
- $p(x)=-2.5x^5+0.7x^2$
  $q(x)=1.2x^4-0.4x^2+3.6$
  $s(x)=p(x)+q(x)=-2.5x^5+1.2x^4+0.3x^2+3.6$
  Write a C++ function with two pointers to polynomial $p(x)$ und $q(x)$ as parameters, which shall add both polynomials and return the resulting polynomial $s(x)=p(x)+q(x)$ as pointer.
  (Hint: the two polynomials $p(x)$ and $q(x)$ do not need to have same degree, i.e. the two lists do not need to have same length.)
  Example:
  $p(x)=-2.5x^5+0.7x^2$
  $q(x)=1.2x^4-0.4x^2+3.6$
  $s(x)=p(x)+q(x)=-2.5x^5+1.2x^4+0.3x^2+3.6$
  further examples see below.
- Write a C++ function **`main`** and
  - input two polynomials $p(x)$ and $q(x)$ by two appropriate function calls.
  - add both polynomials to resulting polynomial $s(x)$ by an appropriate function call.
  - then output all three polynomials in textual form by three appropriate function calls.
  - following input two interval borders **`a`** and **`b`** as well as a step width **`h`** and output a table of value for each of the three polynomials by giving three appropriate function calls.

```
input polynom p:
degree of polynom: 3
coefficient a_3: 1
coefficient a_2: 0
coefficient a_1: 0
coefficient a_0: 0
input polynom q:
degree of polynom: 0
coefficient a_0: 5

p(x)=1*x^3
q(x)=5
s(x)=1*x^3+5

left interval border a for table of values: -5
right interval border b for table of values: 5
step width h for table of values: 2.5

x       p(x)
-5      -125
-2.5    -15.625
0       0
2.5     15.625
5       125

x       q(x)
-5      5
-2.5    5
0       5
2.5     5
5       5

x       s(x)
-5      -120
-2.5    -10.625
0       5
2.5     20.625
5       130
```