In this project we created a two main class, which called "Polynom" and "Monom" that implements several interfaces. And also a "Test" class.

We chose linkedlist as our data Structure, because we think it more useful and dynamic to changes, in a way that saves space in memory. so that in each node will be a one monom.

Note: We assumed that the input is normal , according to Elizabeth.

For ex: 3\*x^2+5 2\*x^2-3

# **Monom**

We needed to do a class which representing a function of the form f(x)= a\*x^b. This class was very useful for us, because we used her methods to the polynom class.

So this class contains constructors and Methods.

## Constructors

1. Get two numbers, first for the coefficient and other for the power, to create a new monom
2. Copy from other monom.
3. Makin from a string.

## Main Methods.

1. [**derivative**](file:///C:\Users\%D7%A0%D7%91%D7%99%D7%90%D7%99\Desktop\%D7%90%D7%9C%D7%A2%D7%93%20%D7%9C%D7%99%D7%9E%D7%95%D7%93%D7%99%D7%9D\Ex0\doc\myMath\Monom.html#derivative--)

This function compute a new monom which is the derivative of this monom.

1. [**f**](file:///C:\Users\%D7%A0%D7%91%D7%99%D7%90%D7%99\Desktop\%D7%90%D7%9C%D7%A2%D7%93%20%D7%9C%D7%99%D7%9E%D7%95%D7%93%D7%99%D7%9D\Ex0\doc\myMath\Monom.html#f-double-)

This functiom compute the value of f(x).

1. [**multiply**](file:///C:\Users\%D7%A0%D7%91%D7%99%D7%90%D7%99\Desktop\%D7%90%D7%9C%D7%A2%D7%93%20%D7%9C%D7%99%D7%9E%D7%95%D7%93%D7%99%D7%9D\Ex0\doc\myMath\Monom.html#multiply-myMath.Monom-)

This function multiplies one monom in another monom.

1. ToString

This function return a string of the monom.

# **Polynom**

A polynomial consists of monoms, as we said earlier, his methods were helped by the Monom functions.

## Constructors

1. Default

This is a default constructor for the polynom. We created a default monom and insert to the Polynom.

1. Copy

This is a copy constructor from one polynom to another.

1. String Transformer

This is a constructor that transforms a string to a polynom.

## Main Methods.

1. Add – to add two polynoms.
2. Copy – from one to another
3. Derivate - Compute a new Polynom which is the derivative of this Polynom.
4. Equals- Test if this Polynom is logically equals to another
5. Multiply- Multiply two polynoms.
6. Root
7. Area

We learned a lot of the efficiency and importance of interfaces, in OOP in java. And we'll try to use them ,If necessary, in the future.

Yoav and Elad.