**Lecture 8 - Advanced parallel algorithms**

**Advanced recursive decomposition**

Sometimes, recursive decomposition can be done more efficiently in a way in which the parts are not straightforward, but reduce the number of operations.

Basic example of such decomposition: compute a complex product by using only 3 (instead of 4) real multiplications).

Solution: (a+bi)\*(c+di) = (ac-bd) + (ad+bc)i = (ac-bd) + (ac+ad+bc+bd-ac-bd)i = (ac-bd) + ((a+b)\*(c+d) - ac - bd)i, which can be computed using only 3 real multiplications (but more additions/substractions).

**Polynomial multiplication using Karatsuba algorithm**

Note: for the classical algorithm, computing the coefficients leads to computing the products and then add diagonals in the following table:  
A black and white diagram of a rectangle

Description automatically generated

Idea:

* Split each of the input polynomials in half;
* Instead of multiplying each of the 4 pairs from the step above, use a similar trick to make only 3 multiplications

Assume input polynomials P(X) and Q(X) of degree 2\*n-1.

Write them as P(X) = P1(X)\*X^n+P2(X) and Q(X) = Q1(X)\*X^n+Q2(X).

Now P(X)\*Q(X) = (P1(X)\*X^n+P2(X)) \* (Q1(X)\*X^n+Q2(X)) =  
= P1(X)\* Q1(X)\*X^2n + (P1(X)\*Q2(X)+P2(X)\*Q1(X))\*X^n + P2(X)\*Q2(X)

But the second term can be written as  
(P1(X)+P2(X)) \* (Q1(X)+Q2(X)) - 1(X)\* Q1(X) - P2(X)\*Q2(X)