Application III – Operations on Polynomials

- Let us consider multiplication
- Can be done as repeated addition.
- So, multiply P1 with each term of P2.
- Add the resulting polynomials.

Develop the pseudocode in class...



Application III – Operations on Polynomials

- We have P1 and P2 arranged in a linked list in decreasing order of exponents.
- We can scan these and add like terms.
 - Need to store the resulting term only if it has non-zero coefficient.
- The number of terms in the result polynomial P1+P2 need not be known in advance.
- We'll use as much space as there are terms in P1+P2.



- Consider another problem described as follows.
- The multiplication of two matrices A and B is understood as follows.
- For each i and j, $C[i,j] = \Sigma_k A[i,k].B[k,j]$.



 If A and B are sparse, there are several issues in matrix multiplication if A, B, and C are stored as arrays.

Storage /Retrieval, Compatibility of indices

· Alternate storage models for sparse matrices exist.

								1	0	2	Α
Row	Col	Val	Row	Col	Val			0	0	0	
1	2	10	1	1	2			U	U	U	
1	3	12	1	2	5			2	5	0	
2	1	1	2	2	1			0	1	0	В
2	3	41:25	1 1/2	<u>դ 1 •••</u>	8 ©	සී	Request contro	ol B	0	0	



0	10	12
1	0	2
0	0	0



2	5	0
0	1	0
8	0	0



96	10	0
18	5	0
0	0	0







- To multiply A and B, get each row of A and each column of B multiply element-wise and sum to get one element of C.
- Not easy if sparse matrix are stored as sorted list. Can we do it efficiently?

Row	Col	Val	Row	Col	Val			0	0
1	2	10	1	1	2				
1	3	12	1	3	8			2	0
2	1	1	2	1	5			5	1
2	3	47:22	% %	₽ 2	🔥	E #	Request contr	ol)	~ 0





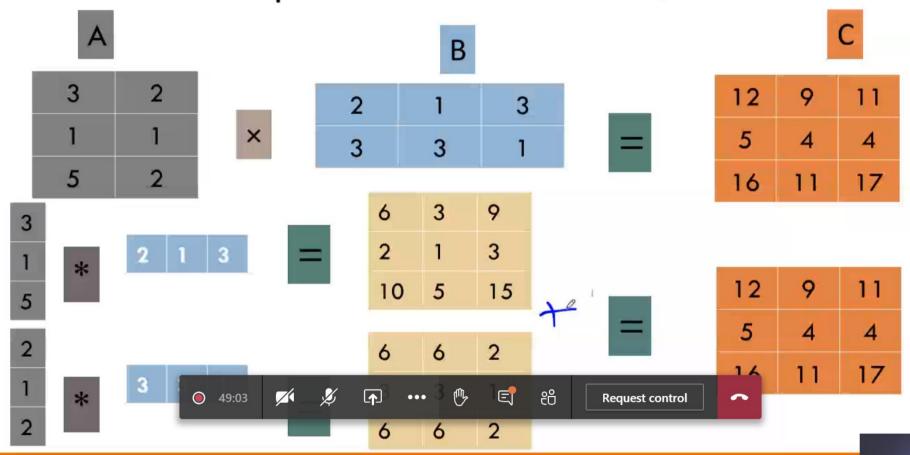
BT

12

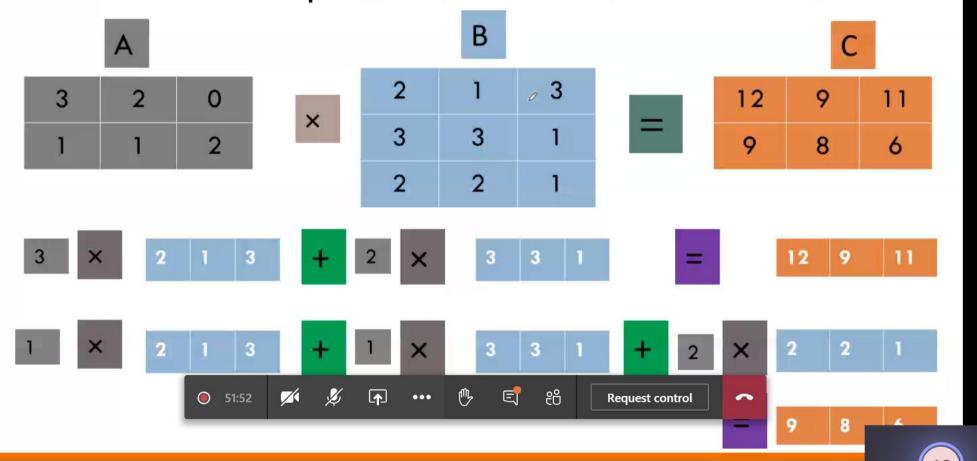
8

0

Matrix Multiplication – Column-Row Formulation



Matrix Multiplication – Row-Row Formulation



Advanced Data Structures & Algorithms