



MONASH University

Information Technology

FIT3143 - LECTURE WEEK 9b

PARALLEL ALGORITHM DESIGN - MATRIX MULTIPLICATION USING
FOX & CANNON WITH VIRTUAL TOPOLOGIES & COLLECTIVE
COMMUNICATIONS

Assignment Project Exam Help

<https://powcoder.com>

algorithm distributed systems database
systems computation knowledge ma
design e-business model data mining int
distributed systems database software
computation knowledge management an

Add WeChat powcoder

Topic Overview

- Quick recap of the matrix multiplication algorithm
- Fox algorithm for parallel matrix multiplication
- Cannon algorithm for parallel matrix multiplication

Assignment Project Exam Help

<https://powcoder.com>

Learning outcome(s) related to this topic

Add WeChat powcoder

- Design and develop parallel algorithms for various parallel computing architectures (LO3)

*These slides and enclosed sample code files were prepared by
Shageenderan Sapai, PhD postgraduate student, Monash University.*

Quick recap of the matrix multiplication algorithm

<https://powcoder.com>

Add WeChat powcoder

Matrix multiplication

Assume we have 2 $n \times n$ matrices, A and B, and we want to find C such that $C = A \times B$

Assignment Project Exam Help
<https://powcoder.com>
Add WeChat powcoder

1	4	8
2	5	6
1	5	6

A

X

7	3	4
2	9	6
7	1	9

B

=

71	47	100
66	57	92
59	54	88

C

$$c_{ij} = (a_i, b_j^T) = \sum_{k=0}^{n-1} a_{ik} \cdot b_{kj}, \quad 0 \leq i < m, \quad 0 \leq j < l$$

Matrix Multiplication

Assignment Project Exam Help
<https://www.powcoder.com>
Add Chat powcoder

1	4	8
2	5	6
1	5	6

A

\times

7	3	4
2	9	6
7	1	9

B

$=$

71		

C

Matrix Multiplication

Assignment Project Exam Help
<https://powcoder.com>
Add Website to powcoder

$$\begin{matrix} \begin{matrix} 1 & 4 & 8 \\ 2 & 5 & 6 \\ 1 & 5 & 6 \end{matrix} & \times & \begin{matrix} 7 & 3 & 4 \\ 2 & 9 & 6 \\ 7 & 1 & 9 \end{matrix} & = & \begin{matrix} 71 & 47 & \\ & & \\ & & \end{matrix} \\ A & & B & & C \end{matrix}$$

Matrix Multiplication

Assignment Project Exam Help
<https://powercoder.com>
Add WeChat : powercoder

1	4	8
2	5	6
1	5	6

A

X

7	3	4
2	9	6
1	9	6

B

=

71	47	100

C

Matrix Multiplication

Assignment Project Exam Help
<https://www.powcoder.com>
Add Chat powcoder

1	4	8
2	5	6
1	5	6

A

\times

7	3	4
2	9	6
7	1	9

B

$=$

71	47	100
66		

C

Matrix Multiplication

Assignment Project Exam Help
<https://powercoder.com>
Add WeChat: powercoder

1	4	8
2	5	6
1	5	6

A

X

7	3	4
2	9	6
7	1	3

B

=

71	47	100
66	57	92
59	54	88

C

Serial Algorithm

A and B nxn matrices => $C=AxB$

```
void Serial_mat_mult(mt_A,mt_B,mt_C,int n)
```

```
{  
    int i,j,k;  
    for (i=0;i<n;i++){  
        for (j=0;j<n;j++){  
            C[i][j]=0.0;  
            for (k=0;k<n;k++)  
                C[i][j]= C[i][j]+A[i][k]*B[k][j];  
        }  
    }  
}
```

Assignment Project Exam Help

<https://powcoder.com>

Add WeChat powcoder

Time Complexity $O(n^3)$ – very slow

Parallel matrix multiplication algorithm - Fox

<https://powcoder.com>

Add WeChat powcoder

Parallel Matrix Multiplication Algorithm - General

A and B nxn matrices => $C = A \times B$

- First, Matrix A & B are partitioned so that each process works on a specific submatrix of A & B.
- Each process does a serial matrix multiplication on their local submatrices of A & B and the results are stored in an accumulating sum
- Local submatrices are then communicated around to calculate the final result
- There are different algorithms for how submatrices are communicated, mainly, **Fox algorithm** and **Cannon algorithm**

Assignment Project Exam Help

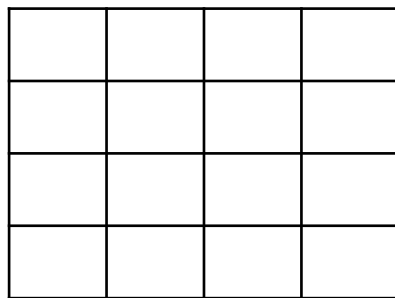
<https://powcoder.com>

Add WeChat powcoder

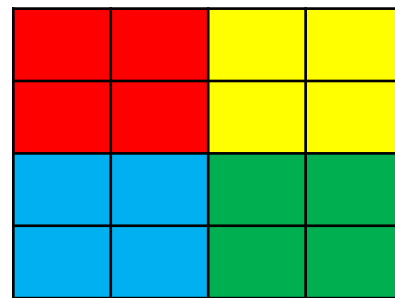
Parallel Algorithm – Data Partitioning

- Parallel matrix multiplication requires data partitioning – common methods are Block-Row Partitioning or Block-Column Partitioning.
- Another method is grid or checkerboard partitioning – which is used in Fox and Cannon algorithm
- Say we have p number of processes and a square Matrix, M , of size $n \times n$ and let $q = \sqrt{p}$
- M is partitioned into p square blocks (sub matrices) $M_{i,j}$ where $0 \leq i, j < q$ of size $n/q \times n/q$

Say $p = 4$ and $n = 4$ Each sub matrix $n/q = 4/\sqrt{4} = 2$



M



$M_{0,0}$	Rank 0
$M_{0,1}$	Rank 1
$M_{1,0}$	Rank 2
$M_{1,1}$	Rank 3

Fox Algorithm

- Matrix A & B are **grid partitioned** into submatrices $A_{i,j}$ and $B_{i,j}$
- In each step, a one-to-all broadcast of the rows of matrix A and a single-step circular upwards shift of columns in matrix B is done

Assignment Project Exam Help

- Loop q times
 1. Broadcast diagonal block $A_{i,i}$ to all processors in the same row group
 2. Multiply block of received $A_{i,i}$ with local block $B_{i,j}$ and add to $C_{i,j}$
 3. Send (shift) entire local block B up within the same column group
 4. Select block $A_{i,(j+1) \bmod q}$ (where $A_{i,j}$ is the block broadcast in the previous step) and broadcast to all processors in the same row group.
 5. Go to 2.
- Gather all

<https://powcoder.com>

Add WeChat powcoder

Data Partitioning – Fox Algorithm

2	4	3	6
9	5	5	7
1	2	8	4
6	9	1	3

A

7	3	4	5
2	9	6	4
7	1	9	2
7	6	4	1

B

0	0	0	0
0	0	0	0
0	0	0	0
0	0	0	0

C

0

2	4
9	5

7	3
2	9

1

3	6
5	7

4	5
6	4

2

1	2
6	9

7	1
7	6

3

8	4
1	3

9	2
4	1

Groups	Processors
Row Group	[0,1] [2,3]
Column Group	[0,2] [1,3]

Loop q times

- Broadcast diagonal block $A_{i,i}$ to all processors in the same row group

$A_{0,0}$

2	4	3	6
9	5	5	7
1	2	8	4
6	9	1	3

A

7	3	4	5
2	9	6	4
7	1	9	2
6	9	1	3

B

0	0	0	0
0	0	0	0
0	0	0	0
0	0	0	0

C

Assignment Project Exam Help

<https://powcoder.com>

Add WeChat powcoder

0

1

2

3

$A_{0,0}$

2	4
9	5

$A_{0,1}$

3	6
5	7

$A_{1,0}$

1	2
6	9

$A_{1,1}$

8	4
1	3

0	0
0	0

7	3
2	9

0	0
0	0

4	5
6	4

0	0
0	0

7	1
7	6

0	0
0	0

9	2
4	1

$A_{i,i}$

$B_{0,0}$

$A_{i,i}$

$B_{0,1}$

$A_{i,i}$

$B_{1,0}$

$A_{i,i}$

$B_{1,1}$

Loop q times

- Broadcast diagonal block $A_{i,i}$ to all processors in the same row group

2	4	3	6
9	5	5	7
1	2	8	4
6	9	1	3

A

7	3	4	5
2	9	6	4
7	1	9	2
6	9	1	3

B

0	0	0	0
0	0	0	0
0	0	0	0
0	0	0	0

C

Assignment Project Exam Help

<https://powcoder.com>

Add WeChat powcoder

0

1

2

3

$A_{0,0}$

$A_{0,1}$

$A_{1,0}$

$A_{1,1}$

2	4
9	5

3	6
5	7

1	2
6	9

8	4
1	3

2	4
9	5

7	3
2	9

0	0
0	0

4	5
6	4

0	0
0	0

7	1
7	6

0	0
0	0

9	2
4	1

$A_{0,0}$

$B_{0,0}$

$A_{i,i}$

$B_{0,1}$

$A_{i,i}$

$B_{1,0}$

$A_{i,i}$

$B_{1,1}$

Loop q times

- Broadcast diagonal block $A_{i,i}$ to all processors in the same row group

2	4	3	6
9	5	5	7
1	2	8	4
6	9	1	3

7	3	4	5
2	9	6	4
7	1	9	2
7	6	1	9

0	0	0	0
0	0	0	0
0	0	0	0
0	0	0	0

Assignment Project Exam Help

<https://powcoder.com>

Add WeChat powcoder

0

1

2

3

$A_{0,0}$

$A_{0,1}$

$A_{1,0}$

$A_{1,1}$

2	4
9	5

3	6
5	7

1	2
6	9

8	4
1	3

2	4
9	5

7	3
2	9

2	4
9	5

4	5
6	4

0	0
0	0

7	1
7	6

0	0
0	0

9	2
4	1

$A_{0,0}$

$B_{0,0}$

$A_{0,0}$

$B_{0,1}$

$A_{i,i}$

$B_{1,0}$

$A_{i,i}$

$B_{1,1}$

Loop q times

- Broadcast diagonal block $A_{i,i}$ to all processors in the same row group

2	4	3	6
9	5	5	7
1	2	8	4
6	9	1	3

A

7	3	4	5
2	9	6	4
7	1	9	2
6	9	1	3

B

0	0	0	0
0	0	0	0
0	0	0	0
0	0	0	0

C

Assignment Project Exam Help

<https://powcoder.com>

Add WeChat powcoder

0

1

2

3

$A_{0,0}$

2	4
9	5

$A_{0,1}$

3	6
5	7

$A_{1,0}$

1	2
6	9

$A_{1,1}$

8	4
1	3

2	4
9	5

$A_{0,0}$

7	3
2	9

$B_{0,0}$

2	4
9	5

$A_{0,0}$

4	5
6	4

$B_{0,1}$

8	4
1	3

$A_{1,1}$

7	1
7	6

$B_{1,0}$

0	0
0	0

$A_{i,i}$

9	2
4	1

$B_{1,1}$

Loop q times

- Broadcast diagonal block $A_{i,i}$ to all processors in the same row group

2	4	3	6
9	5	5	7
1	2	8	4
6	9	1	3

A

7	3	4	5
2	9	6	4
7	1	9	2
7	1	9	2

B

0	0	0	0
0	0	0	0
0	0	0	0
0	0	0	0

C

0

1

2

3

$A_{0,0}$

2	4
9	5

$A_{0,1}$

3	6
5	7

$A_{1,0}$

1	2
6	9

$A_{1,1}$

8	4
1	3

2	4
9	5

$A_{0,0}$

7	3
2	9

$B_{0,0}$

2	4
9	5

$A_{0,0}$

4	5
6	4

$B_{0,1}$

8	4
1	3

$A_{1,1}$

7	1
7	6

$B_{1,0}$

8	4
1	3

$A_{1,1}$

9	2
4	1

$B_{1,1}$

Loop q times

- Multiply block of received A with resident block B and add to C

2	4	3	6
9	5	5	7
1	2	8	4
6	9	1	3

A

7	3	4	5
2	9	6	4
7	1	9	2
6	6	9	2

B

22	42	32	26
73	72	66	65
84	32	88	20
28	19	21	5

C

Assignment Project Exam Help

<https://powcoder.com>

Add WeChat powcoder

0

1

2

3

$A_{0,0}$

2	4
9	5

$A_{0,1}$

3	6
5	7

$A_{1,0}$

1	2
6	9

$A_{1,1}$

8	4
1	3

2	4
9	5

 \times

7	3
2	9

$A_{0,0}$

$B_{0,0}$

2	4
9	5

 \times

4	5
6	4

$A_{0,0}$

$B_{0,1}$

8	4
1	3

 \times

7	1
7	6

$A_{1,1}$

$B_{1,0}$

8	4
1	3

 \times

9	2
4	1

$A_{1,1}$

$B_{1,1}$

Loop q times

- Multiply block of received A with resident block B and add to C

2	4	3	6
9	5	5	7
1	2	8	4
6	9	1	3

A

7	3	4	5
2	9	6	4
7	1	9	2
7	6	1	9

B

22	42	32	26
73	72	66	65
84	32	88	20
29	19	21	5

C

Assignment Project Exam Help

<https://powcoder.com>

Add WeChat powcoder

0

$A_{0,0}$

2	4
9	5

1

$A_{0,1}$

3	6
5	7

2

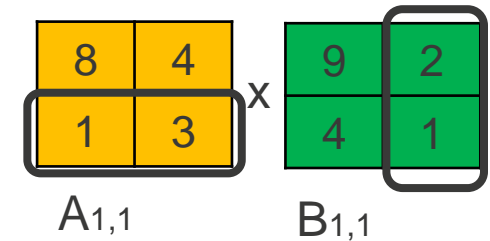
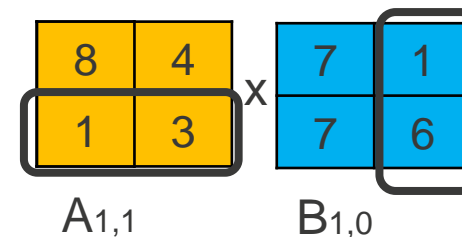
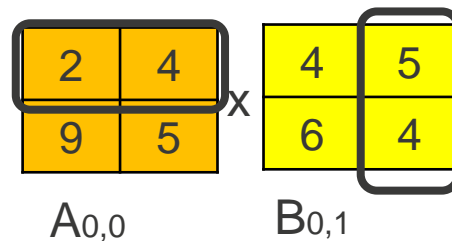
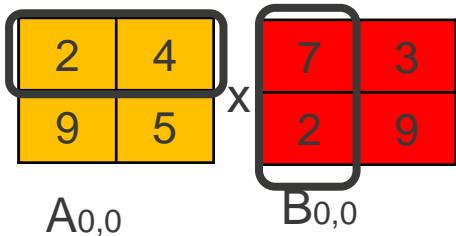
$A_{1,0}$

1	2
6	9

3

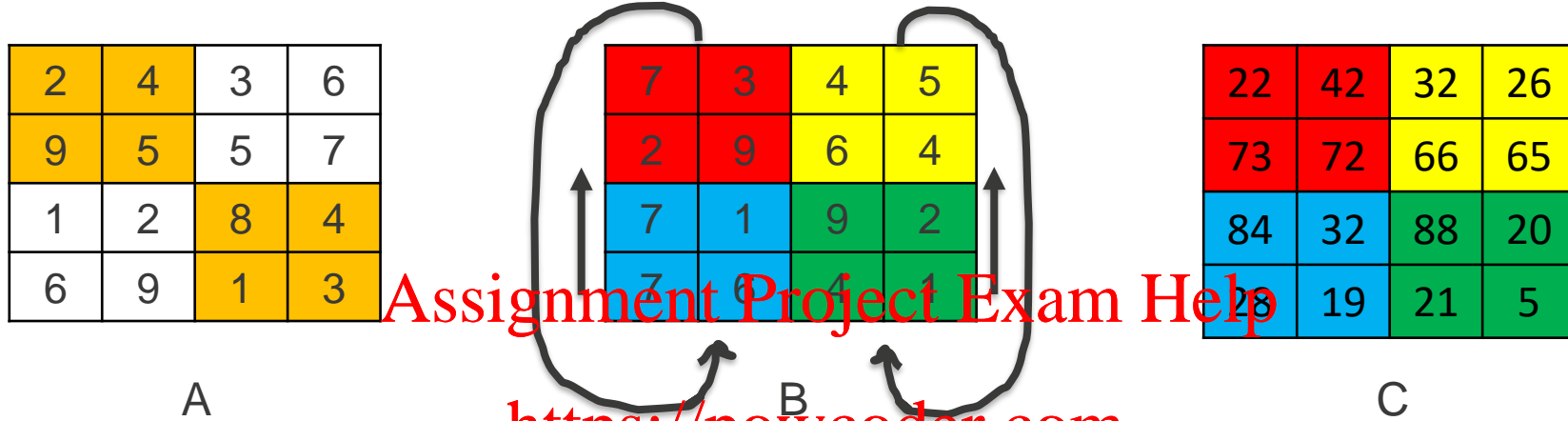
$A_{1,1}$

8	4
1	3

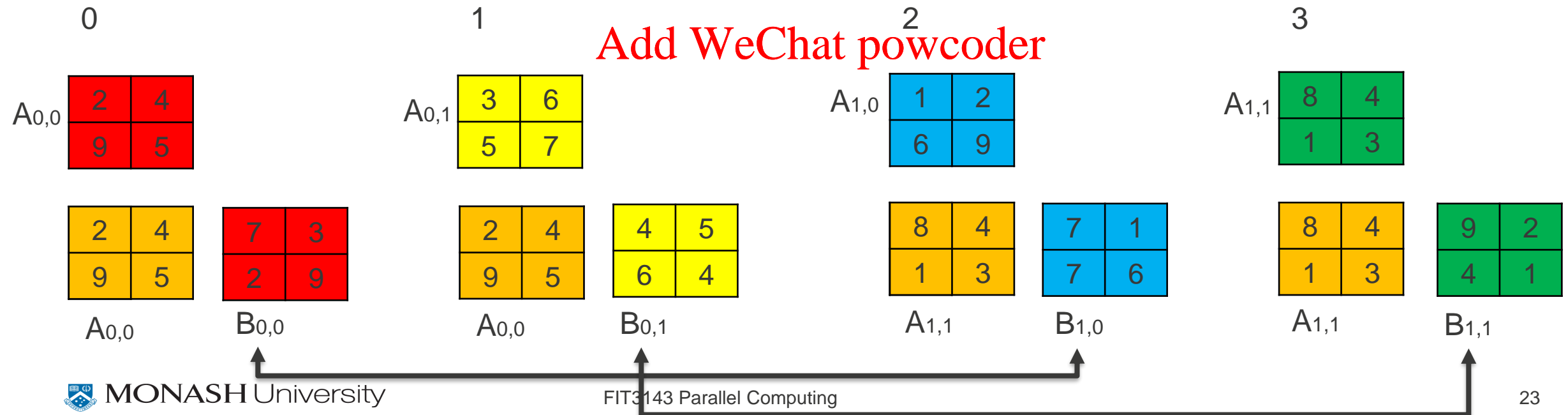


Loop q times

1. Send (shift) entire local block B up within the same column group



Add WeChat powcoder



Loop q times

- Send (shift) **entire** resident block B up one step

2	4	3	6
9	5	5	7
1	2	8	4
6	9	1	3

A

7	1	9	2
7	6	4	1
7	3	4	5
2	9	6	4

B

22	42	32	26
73	72	66	65
84	32	88	20
28	19	21	5

C

Assignment Project Exam Help

<https://powcoder.com>

Add WeChat powcoder

0

A_{0,0}

2	4
9	5

1

A_{0,1}

3	6
5	7

2

A_{1,0}

1	2
6	9

3

A_{1,1}

8	4
1	3

2	4
9	5

A_{0,0}

7	1
7	6

B_{0,0}

2	4
9	5

A_{0,0}

9	2
4	1

B_{0,1}

8	4
1	3

A_{1,1}

7	3
2	9

B_{1,0}

8	4
1	3

A_{1,1}

4	5
6	4

B_{1,1}

Loop q times

- Broadcast $A_{i,(j+1) \bmod q}$ to all processors in the same row group

	2	4	3	6	
	9	5	5	7	$A_{0,1}$
$A_{1,0}$	1	2	8	4	
	6	9	1	3	

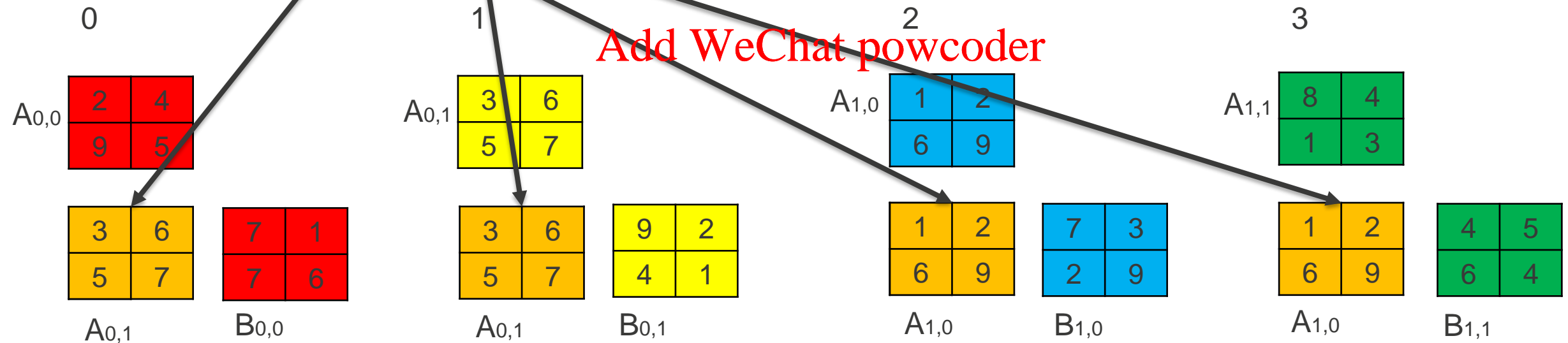
7	1	9	2
7	6	4	1
7	3	4	5
2	9	6	4

22	42	32	26
73	72	66	65
84	32	88	20
28	19	21	5

Assignment Project Exam Help

<https://powcoder.com>

Add WeChat powcoder



Loop q times

- Multiply block of received A with resident block B and add to C

2	4	3	6
9	5	5	7
1	2	8	4
6	9	1	3

A

7	1	9	2
7	6	4	1
7	3	4	5
2	9	6	4

B

85	81	83	38
157	119	139	82
95	53	104	33
88	118	99	71

C

Assignment Project Exam Help

<https://powcoder.com>

Add WeChat powcoder

0

$A_{0,0}$

2	4
9	5

1

$A_{0,1}$

3	6
5	7

2

$A_{1,0}$

1	2
6	9

3

$A_{1,1}$

8	4
1	3

3	6
5	7

x

7	1
7	6

$A_{0,1}$

$B_{0,0}$

3	6
5	7

x

9	2
4	1

$A_{0,1}$

$B_{0,1}$

1	2
6	9

x

7	3
2	9

$A_{1,0}$

$B_{1,0}$

1	2
6	9

x

4	5
6	4

$A_{1,0}$

$B_{1,1}$

Loop q times

- Multiply block of received A with resident block B and add to C

2	4	3	6
9	5	5	7
1	2	8	4
6	9	1	3

A

7	1	9	2
7	6	4	1
7	3	4	5
2	9	6	4

B

85	81	83	38
157	119	139	82
95	53	104	33
88	118	99	71

C

Assignment Project Exam Help

<https://powcoder.com>

Add WeChat powcoder

0

$A_{0,0}$

2	4
9	5

1

$A_{0,1}$

3	6
5	7

2

$A_{1,0}$

1	2
6	9

3

$A_{1,1}$

8	4
1	3

3	6
5	7

$A_{0,1}$

x

7	1
7	6

$B_{0,0}$

$A_{0,1}$

$B_{0,0}$

3	6
5	7

$A_{0,1}$

x

9	2
4	1

$B_{0,1}$

$A_{0,1}$

$B_{0,1}$

1	2
6	9

$A_{1,0}$

x

7	3
2	9

$B_{1,0}$

$A_{1,0}$

$B_{1,0}$

1	2
6	9

$A_{1,0}$

x

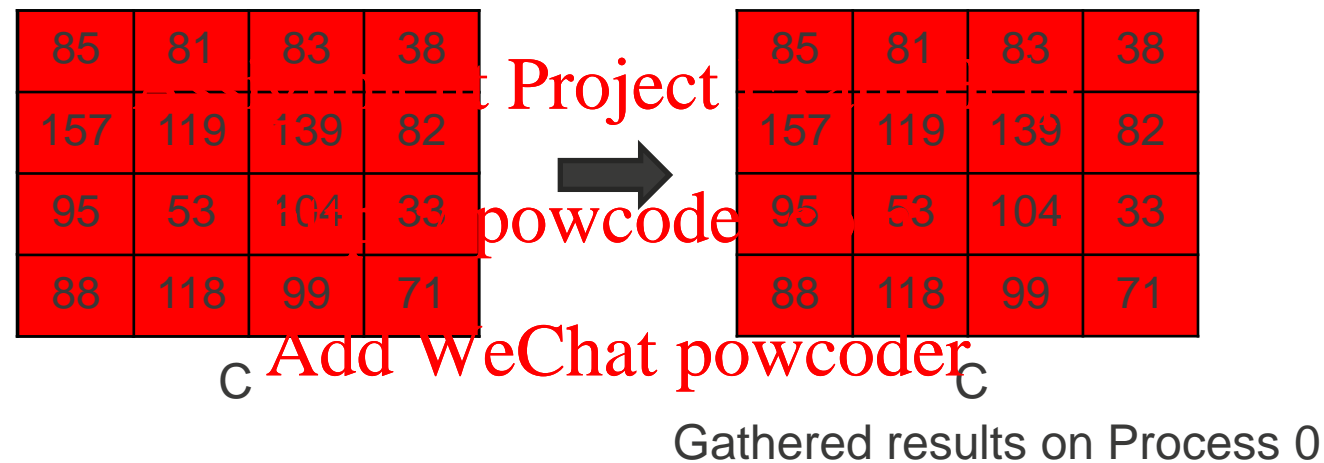
4	5
6	4

$B_{1,1}$

$A_{1,0}$

$B_{1,1}$

- Already looped $q(2)$ times, so now we must:
- Gather All



Fox Algorithm

- Tends to be faster than simple parallel matrix multiplication for large values of n
- Implemented in MPI by creating row and column communicators which enable easy communication of A & B blocks
- Although dealing with 2D matrices, in coding implementation these are represented using 1D arrays
- Difficult to adapt for non-square matrices i.e. $A (m \times n) \times B (n \times m)$
- Has high communication overhead because at each step we are sending (relatively) large blocks of local A & B.

Assignment Project Exam Help

<https://powcoder.com>

Add WeChat powcoder

Parallel matrix multiplication algorithm - Cannon

<https://powcoder.com>

Add WeChat powcoder

Cannon Algorithm

- Matrix A & B are **grid partitioned** into submatrices $A_{i,j}$ and $B_{i,j}$
- Data is moved incrementally in $q-1$ phases (ring broadcast algorithm)
- Each process skews matrix $A_{i,j}$ and $B_{i,j}$ to align elements by shifting $A_{i,j}$ by i rows to the left and shifting $B_{i,j}$ by j columns up.
- Multiply local block A with local block B and add to local C
- Loop $q-1$ times
 1. Shift elements (each A_i row by 1 unit and each B_j column by 1 unit)
 2. Multiply local block A with local block B and add to local C
- Gather all

Assignment Project Exam Help

<https://powcoder.com>

Add WeChat powcoder

Data Partitioning – Cannon Algorithm

2	4	3	6
9	5	5	7
1	2	8	4
6	9	1	3

A

7	3	4	5
2	9	6	4
7	1	9	2
7	6	4	1

B

0	0	0	0
0	0	0	0
0	0	0	0
0	0	0	0

C

0

2	4
9	5

1

3	6
5	7

2

1	2
6	9

3

8	4
1	3

7	3
2	9

4	5
6	4

7	1
7	6

9	2
4	1

Skew matrix A and B to align elements by shifting $A_{i,j}$ by i rows to the left and shift $B_{i,j}$ by j columns up.

2	4	3	6
9	5	5	7
1	2	8	4
6	9	1	3

7	3	4	5
2	9	6	4
7	1	9	2
7	6	4	4

A <https://powcoder.com> B

Add WeChat powcoder

0123

Add WeChat powcoder

Local A

2	4
9	5

3	6
5	7

1	2
6	9

8	4
1	3

Local B

7	3
2	9

4	5
6	4

7	1
7	6

9	2
4	1

Skew matrix A and B to align elements by shifting A_i by i rows to the left and shift B_i by i columns up.

2	4	3	6
9	5	5	7
1	2	8	4
6	9	1	3

7	3	4	5
2	9	6	4
7	1	9	2
7	6	4	4

A

B

<https://powcoder.com>

Add WeChat powcoder

0

1

2

3

Local A

2	4
9	5

3	6
5	7

8	4
1	3

1	2
6	9

Local B

7	3
2	9

9	2
4	1

7	1
7	6

4	5
6	4

Skew matrix A and B to align elements by shifting A_i by i rows to the left and shift B_i by i columns up.

2	4	3	6
9	5	5	7
8	4	1	2
1	3	6	9

7	3	9	2
2	9	4	1
7	1	4	5
7	6	6	4

A

B

<https://powcoder.com>

Add WeChat powcoder

0

1

2

3

Local A

2	4
9	5

3	6
5	7

8	4
1	3

1	2
6	9

Local B

7	3
2	9

9	2
4	1

7	1
7	6

4	5
6	4

Loop q-1 times

1. Multiply elements and add to accumulating sum

2	4	3	6
9	5	5	7
8	4	1	2
1	3	6	9

A

7	3	9	2
2	9	4	1
7	1	4	5
7	6	6	4

B

22	42	51	12
73	72	73	17
84	32	16	13
28	19	78	66

C

Assignment Project Exam Help

<https://powcoder.com>

Add WeChat powcoder

0

2	4
9	5

Local A

x

7	3
2	9

Local B

1

3	6
5	7

Local A

x

9	2
4	1

Local B

2

8	4
1	3

Local A

x

7	1
7	6

Local B

3

1	2
6	9

Local A

x

4	5
6	4

Local B

Loop q1 times

1. Shift elements (each A_i row by 1 unit and each B_i column by 1 unit)

2	4	3	6
9	5	5	7
8	4	1	2
1	3	6	9

7	3	9	2
2	9	4	1
7	1	4	5
7	6	6	4

Assignment Project Exam Help

A

B

<https://powcoder.com>

Add WeChat powcoder

0

1

2

3

Local A

2	4
9	5

3	6
5	7

8	4
1	3

1	2
6	9

Local B

7	3
2	9

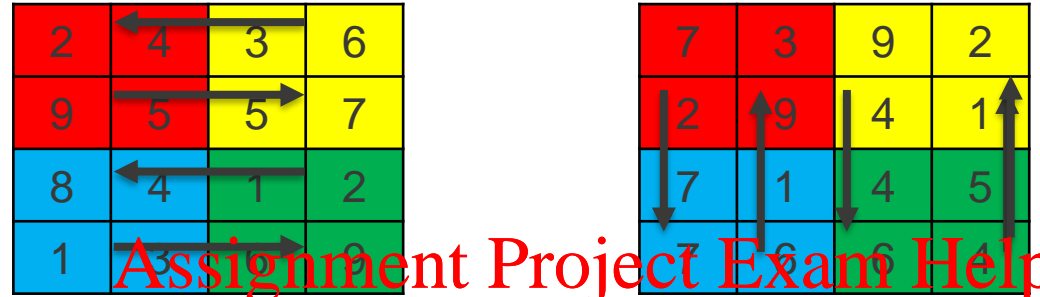
9	2
4	1

7	1
7	6

4	5
6	4

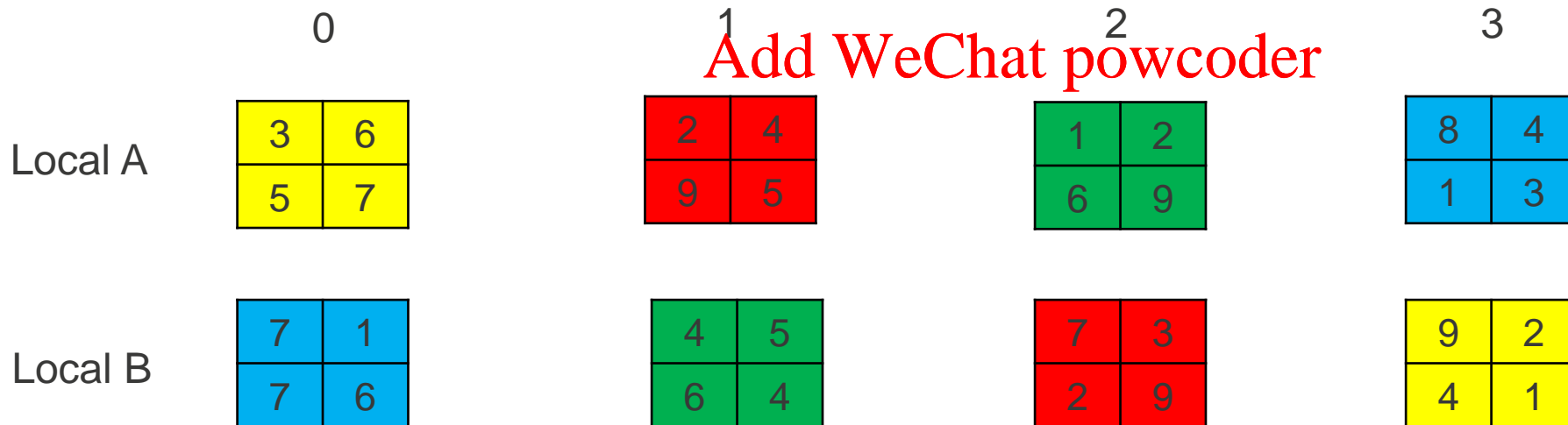
Loop q-1 times

1. Shift elements (each A_i row by 1 unit and each B_i column by 1 unit)



<https://powcoder.com>

Add WeChat powcoder



Loop q-1 times

1. Shift elements (each A_i row by 1 unit and each B_i column by 1 unit)

3	6	2	4
5	7	9	5
1	2	8	4
6	9	1	3

7	1	4	5
7	6	6	4
7	3	9	2
2	9	4	1

Assignment Project Exam Help

A <https://powcoder.com> B

Add WeChat powcoder

	0	1	2	3																
	Add WeChat powcoder																			
Local A	<table><tr><td>3</td><td>6</td></tr><tr><td>5</td><td>7</td></tr></table>	3	6	5	7	<table><tr><td>2</td><td>4</td></tr><tr><td>9</td><td>5</td></tr></table>	2	4	9	5	<table><tr><td>1</td><td>2</td></tr><tr><td>6</td><td>9</td></tr></table>	1	2	6	9	<table><tr><td>8</td><td>4</td></tr><tr><td>1</td><td>3</td></tr></table>	8	4	1	3
3	6																			
5	7																			
2	4																			
9	5																			
1	2																			
6	9																			
8	4																			
1	3																			
Local B	<table><tr><td>7</td><td>1</td></tr><tr><td>7</td><td>6</td></tr></table>	7	1	7	6	<table><tr><td>4</td><td>5</td></tr><tr><td>6</td><td>4</td></tr></table>	4	5	6	4	<table><tr><td>7</td><td>3</td></tr><tr><td>2</td><td>9</td></tr></table>	7	3	2	9	<table><tr><td>9</td><td>2</td></tr><tr><td>4</td><td>1</td></tr></table>	9	2	4	1
7	1																			
7	6																			
4	5																			
6	4																			
7	3																			
2	9																			
9	2																			
4	1																			

Loop q-1 times

1. Multiply local block A with local block B and add to local C

3	6	2	4
5	7	9	5
1	2	8	4
6	9	1	2

A

7	1	4	5
7	6	6	4
7	3	9	2
2	9	4	1

B

22	42	51	12
73	72	73	17
84	32	16	13
28	19	78	66

C

<https://powcoder.com>

Add WeChat powcoder

0

3	6
5	7

Local A

x

7	1
7	6

Local B

1

2	4
9	5

Local A

x

4	5
6	4

Local B

2

1	2
6	9

Local A

x

7	3
2	9

Local B

3

8	4
1	3

Local A

x

9	2
4	1

Local B

Loop q-1 times

1. Multiply local block A with local block B and add to local C

3	6	2	4
5	7	9	5
1	2	8	4
6	9	1	2

A

7	1	4	5
7	6	6	4
7	3	9	2
2	9	4	1

B

85	81	83	38
157	119	139	82
95	53	104	33
88	118	99	71

C

<https://powcoder.com>

Add WeChat powcoder

0

3	6
5	7

x

7	1
7	6

Local A

Local B

1

2	4
9	5

x

4	5
6	4

Local A

Local B

2

1	2
6	9

x

7	3
2	9

Local A

Local B

3

8	4
1	3

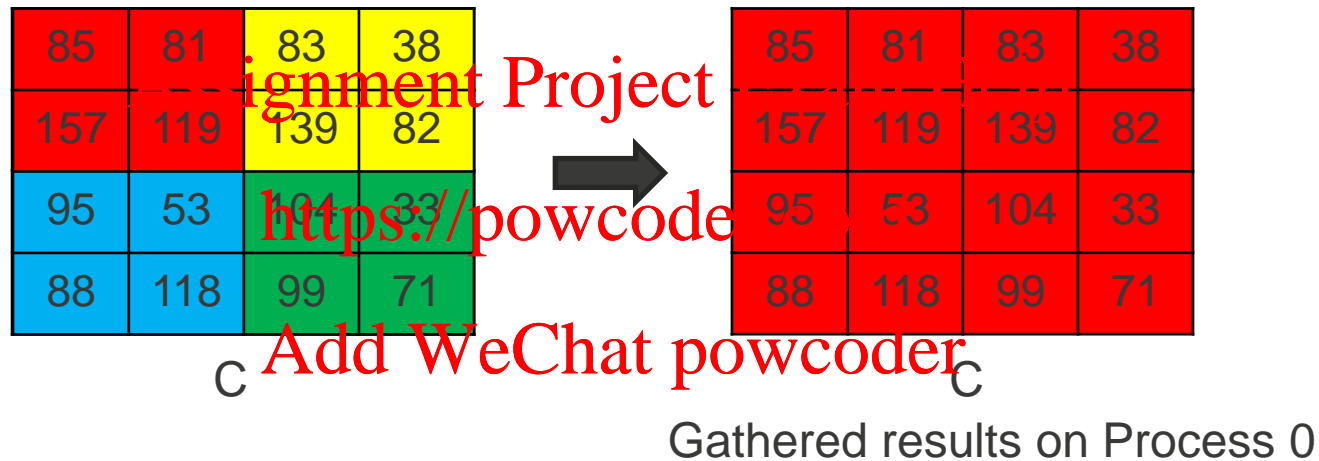
x

9	2
4	1

Local A

Local B

- Already looped $q-1(1)$ times, so now we must:
- Gather All



Cannon Algorithm

- Tends to be faster than simple parallel matrix multiplication for large values of n
- Implemented in MPI by creating row and column communicators which enable easy shifting of data
- Although dealing with 2D matrices, in coding implementation these are represented using 1D arrays
- Difficult to adapt for non-square matrices i.e. $A (m \times n) \times B (n \times m)$
- Has lower communication overhead than Fox algorithm because at each step we are sending less data than in Fox

Assignment Project Exam Help

<https://powcoder.com>

Add WeChat powcoder

Cannon Algorithm – Bigger Example

2	4	3	6	4	4
9	5	5	7	3	3
1	2	8	4	2	2
6	9	1	3	1	1
1	2	8	4	2	2
6	9	1	3	1	1

A

7	3	4	5	4	4
2	9	6	4	5	5
7	6	4	1	7	7
7	6	4	1	7	7
7	6	4	1	7	7
7	6	4	1	7	7

B

0	0	0	0	0	0
0	0	0	0	0	0
0	0	0	0	0	0
0	0	0	0	0	0
0	0	0	0	0	0
0	0	0	0	0	0

C

Assignment Project Exam Help

<https://powcoder.com>

Add WeChat powcoder

Skew

2	4	3	6	4	4
9	5	5	7	3	3
8	4	2	2	1	2
1	3	1	1	6	9
2	2	1	2	8	4
1	1	6	9	1	3

A

7	3	9	2	6	6
2	9	4	1	7	7
7	1	9	2	4	4
7	6	4	1	5	5
7	1	9	2	4	4
7	6	4	1	5	5

B

0	0	0	0	0	0
0	0	0	0	0	0
0	0	0	0	0	0
0	0	0	0	0	0
0	0	0	0	0	0
0	0	0	0	0	0

C

Assignment Project Exam Help

<https://powcoder.com>

Add WeChat powcoder

Multiply

2	4	3	6	4	4
9	5	5	7	3	3
8	4	2	2	1	2
1	3	1	1	6	9
2	2	1	2	8	4
1	1	6	9	1	3

A

7	3	9	2	6	6
2	9	4	1	7	7
7	1	9	2	4	4
7	6	4	1	5	5
7	4	5	6	3	3
7	6	6	4	7	7

B

22	42	51	12	52	52
73	72	73	17	39	39
84	32	26	6	14	14
28	19	13	3	69	69
28	14	16	13	76	76
14	7	78	66	27	27

C

Assignment Project Exam Help

<https://powcoder.com>

Add WeChat powcoder

Skew

3	6	4	4	2	4
5	7	3	3	9	5
2	2	1	2	8	4
1	1	6	9	1	3
1	2	8	4	2	2
6	9	1	3	1	1

A

7	1	9	2	4	4
7	6	4	1	5	5
7	1	4	5	6	6
7	6	6	4	7	7
7	9	2	6	8	8
2	9	4	1	7	7

B

22	42	51	12	52	52
73	72	73	17	39	39
84	32	26	6	14	14
28	19	13	3	69	69
28	14	16	13	76	76
14	7	78	66	27	27

C

Assignment Project Exam Help

<https://powcoder.com>

Add WeChat powcoder

Multiply

3	6	4	4	2	4
5	7	3	3	9	5
2	2	1	2	8	4
1	1	6	9	1	3
1	2	8	4	2	2
6	9	1	3	1	1

A

7	1	9	2	4	4
7	6	4	1	5	5
7	1	4	5	6	6
7	6	6	4	7	7
7	1	9	2	6	6
2	9	4	1	7	7

B

85	81	103	24	80	80
157	119	112	26	100	100
112	46	42	19	90	90
42	46	91	69	96	96
39	35	104	33	102	102
74	106	99	71	40	40

C

Assignment Project Exam Help

<https://powcoder.com>

Add WeChat powcoder

Skew

4	4	2	4	3	6
3	3	9	5	5	7
1	2	8	4	2	2
6	9	1	3	1	1
8	4	2	2	1	2
1	3	1	1	6	9

A

7	1	4	5	6	6
7	6	6	4	7	7
7	3	9	2	6	6
2	9	4	1	7	7
7	1	9	2	4	4
7	6	4	1	5	5

B

85	81	103	24	80	80
157	119	112	26	100	100
112	46	42	19	90	90
42	26	91	69	96	96
39	35	104	33	102	102
74	106	99	71	40	40

C

Assignment Project Exam Help

<https://powcoder.com>

Add WeChat powcoder

Multiply

4	4	2	4	3	6
3	3	9	5	5	7
1	2	8	4	2	2
6	9	1	3	1	1
8	4	2	2	1	2
1	3	1	1	6	9

A

7	1	4	5	6	6
7	6	6	4	7	7
7	3	9	2	6	6
2	9	4	1	7	7
7	1	9	2	4	4
7	6	4	1	5	5

B

141	109	135	50	140	140
199	140	178	91	179	179
123	67	130	39	116	116
102	125	112	74	109	109
123	67	130	39	116	116
102	125	112	74	109	109

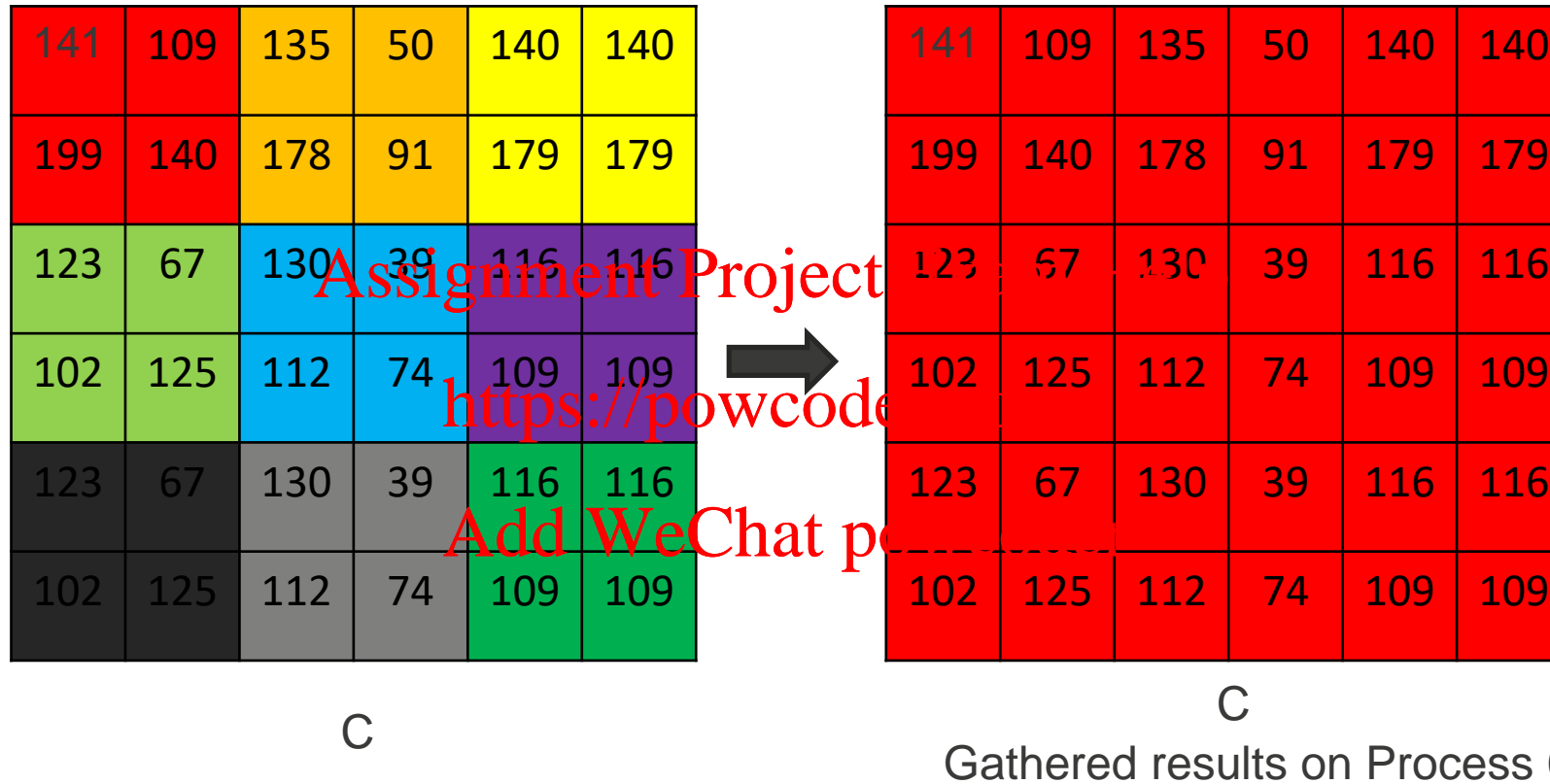
C

Assignment Project Exam Help

<https://powcoder.com>

Add WeChat powcoder

- Already looped $q-1(2)$ times, so now we must:
- Gather All



C code files implementing fox and cannon parallel matrix multiplication with MPI

- Fox – Click [here](#)
 - Cannon – Click [here](#)
- <https://powcoder.com>

Add WeChat powcoder