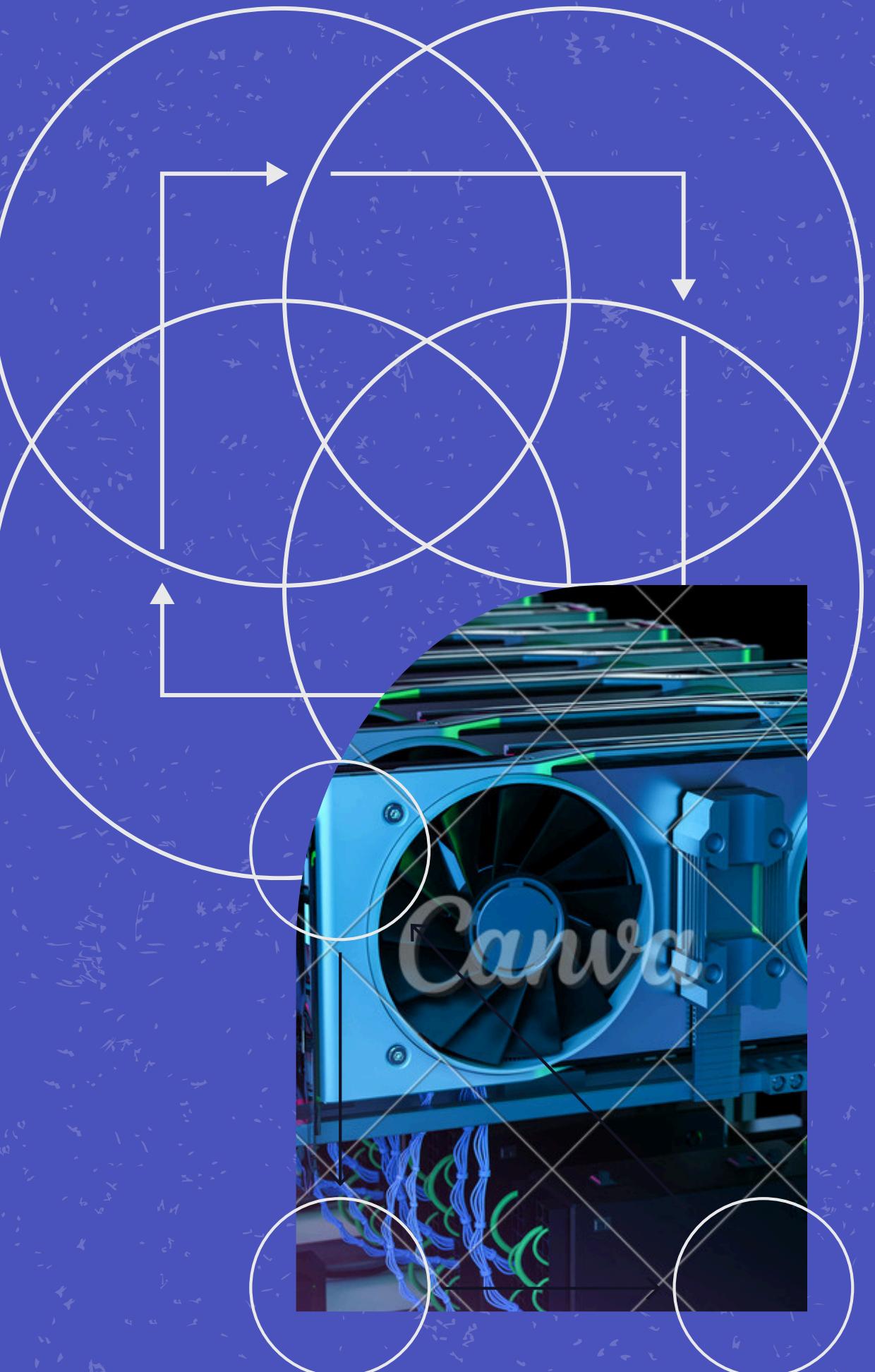


Accelerating Matrix Multiplication

# Strassen Matrix Multiplication

Presented by Yagnych Anastasia



# Understanding the Algorithms



## 01 Classic algorithm

Using classical methods of linear algebra. Easy to implement, but hard to optimize.

Time to multiply -  $O(n^3)$ , very effective for small matrix

## 02 Strassen algorithm

Matrix multiplication tasks can be completed much faster on GPUs than on CPUs, but using a faster algorithm is a better solution. Strassen algorithm reduce the multiplications. Time to complete is  $O(n^{2.81})$

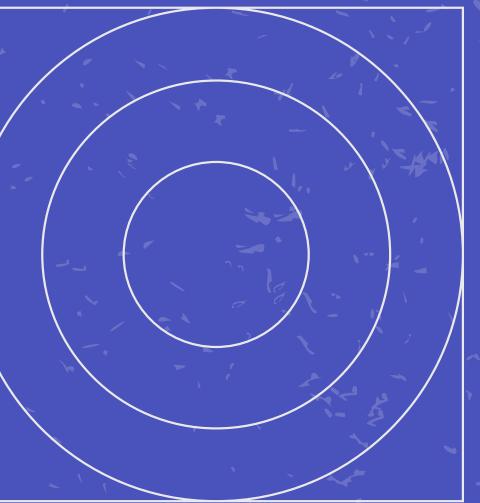
## 03 Why we still using classical algorithm?

Classical matrix multiplication is still widely used because it is more numerically stable and using less memory. Also classic method are better for small and medium matrix

# Code for CUDA:

Using classical algo for slide:

```
__global__ void matMulClassic(float* A,  
float* B, float* C, int N) {  
    int row = blockIdx.y * blockDim.y +  
    threadIdx.y;  
    int col = blockIdx.x * blockDim.x +  
    threadIdx.x;  
    float sum = 0.0f;  
    for (int k = 0; k < N; ++k) {  
        sum += A[row * N + k] * B[k * N + col];  
    }  
    C[row * N + col] = sum;  
}
```



# How Strassen algo work?

## STEP 1

Divide: Split each input  $n \times n$  matrix A and B into four  $n/2 \times n/2$  submatrices.

## STEP 2

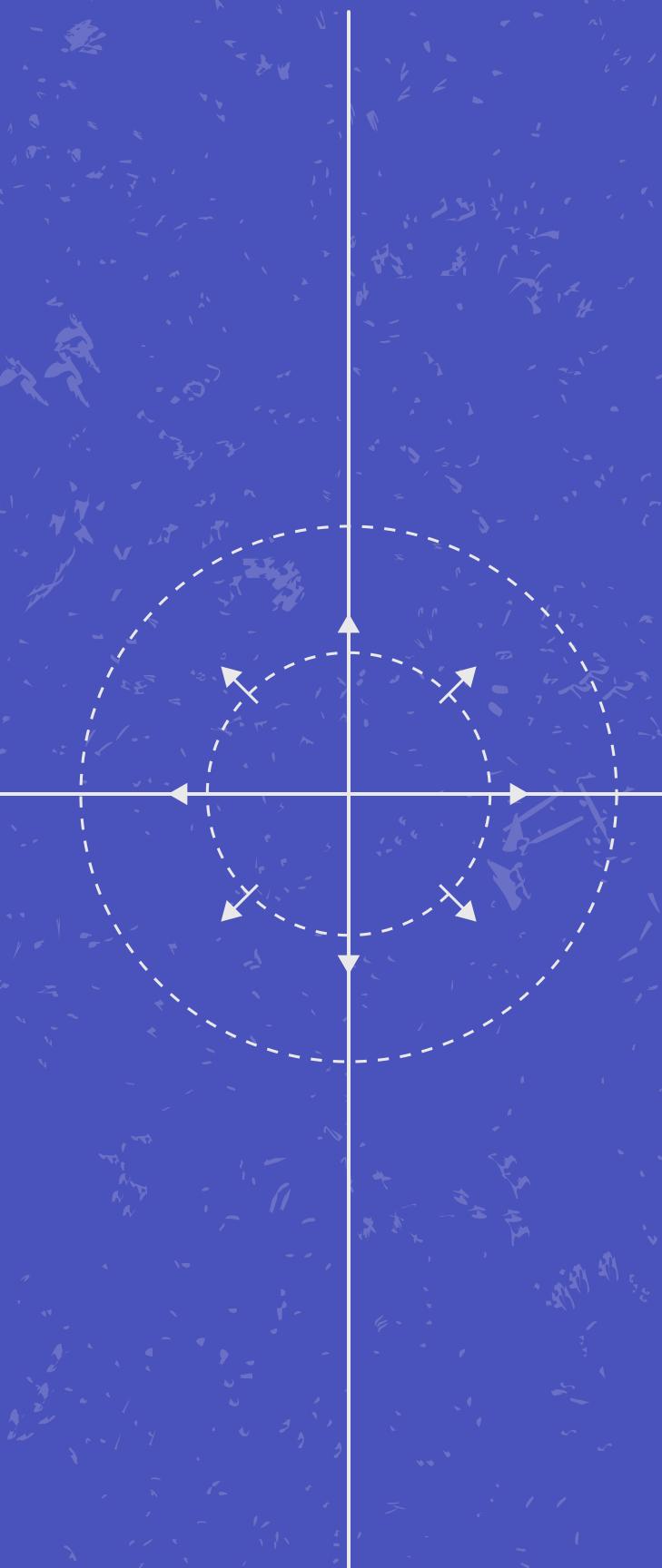
Calculate 7 specific matrix products (instead of 8 in classical)

## STEP 3

Construct the result matrix C submatrices from the 7 products

## STEP 4

Recur or Stop: repeat the process recursively

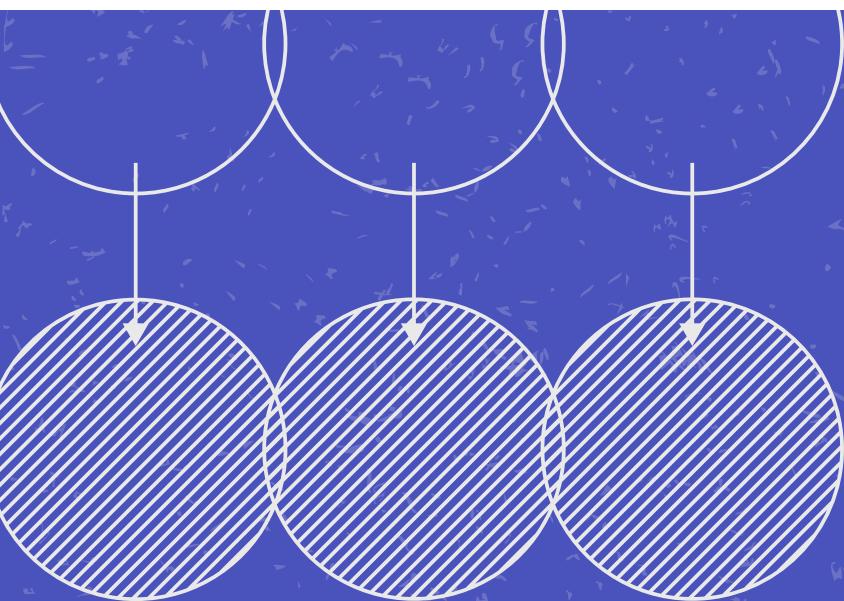
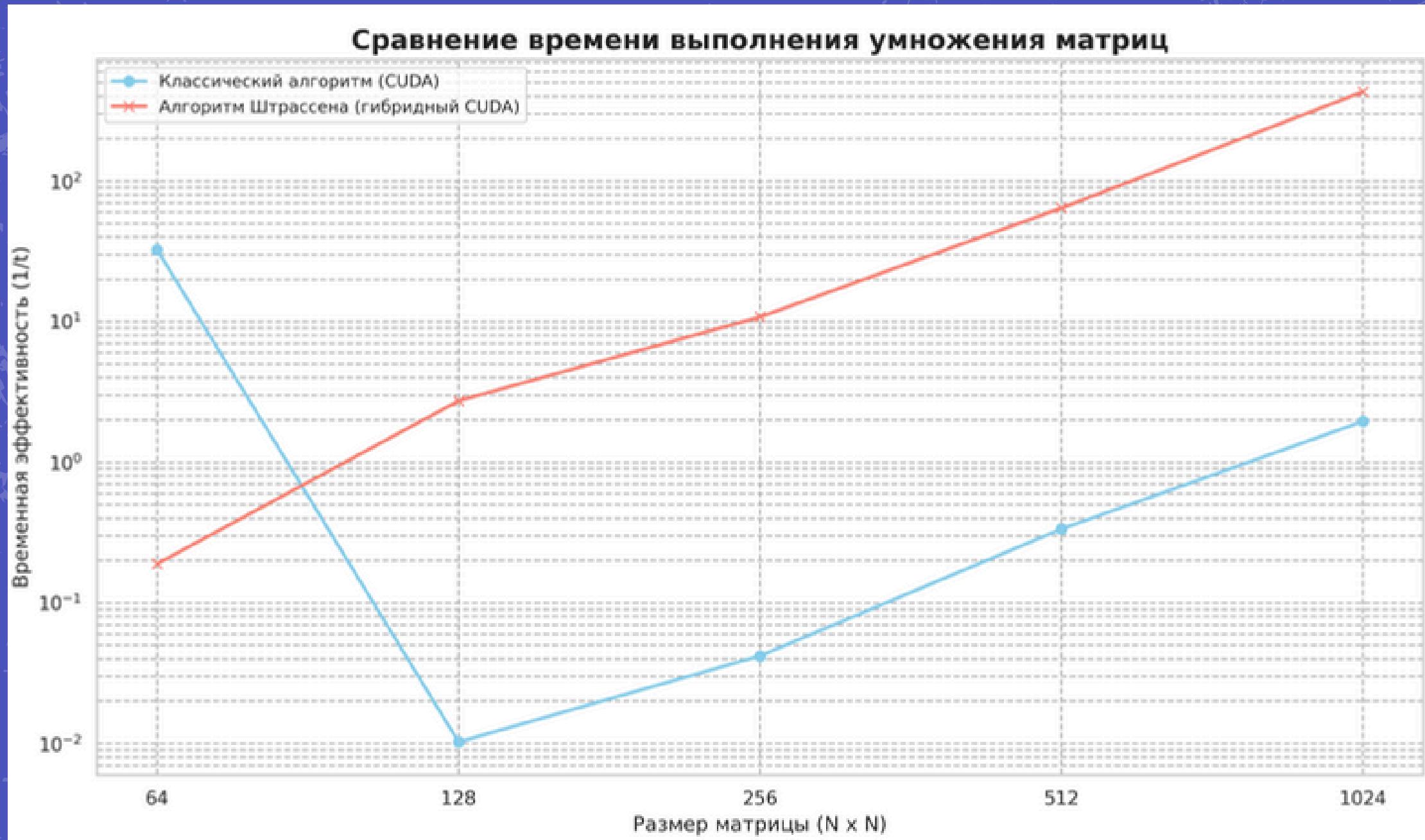


# Validation of Results

How to check what algo work right?

- Compare the trace (sum of diagonal elements) of result matrices.
- Compare matrix fully (could be difficult)

RESULTS OF TESTING STRASSEN VS CLASSICAL



# Summary and Learnings



## 01 GPU optimization

- Successfully implemented two GPU-accelerated matrix algorithms.

## 02 Strassen algorithm

- Demonstrated Strassen's performance gain for big matrix, and similar results for medium

## 03 Coding using CUDA-language

- Gained practical CUDA programming experience.