

Matrix Multiplication Using OpenMP in CPP

- Parallel programming is a technique used to improve the performance of applications by splitting the work into multiple threads or processes that can run simultaneously on different CPUs or CPU cores.
- Matrix multiplication is a fundamental operation in linear algebra, and it involves multiplying two matrices to produce a third matrix.
- The algorithm for matrix multiplication is computationally intensive and can benefit from parallelization.

OpenMP

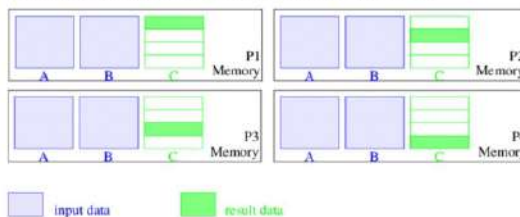
- OpenMP is a popular API for parallel programming in C++, which provides a set of compiler directives, runtime library routines, and environment variables for shared memory parallel programming.
- In this Example, we are using OpenMP to improve the performance of Matrix Multiplication by splitting the work into multiple threads or processes that can run simultaneously on different CPU cores.



Implementaion

Version 1 of Parallel Matrix Multiplication:

Copy matrices A and B to all nodes and compute (index driven) the corresponding fraction of result matrix C concurrently (Example for 4 processors):



The result matrix C will be distributed among the processors in row blocks (the same scheme is possible for column blocks)

Using OpenMP for Matrix Multiplication Calculate executino time

```
1 // Parallel matrix multiplication in parallel using OpenMP
2 auto start_parallel = chrono::high_resolution_clock::now();
3 #pragma omp parallel for
4 for (int i = 0; i < N; i++) {
5     for (int j = 0; j < N; j++) {
6         int sum = 0;
7         for (int k = 0; k < N; k++) {
8             sum += A[i][k] * B[k][j];
9         }
10        C[i][j] = sum;
11    }
12 }
13 auto end_parallel = chrono::high_resolution_clock::now();
14 auto duration_parallel = chrono::duration_cast<chrono::milliseconds>(end_parallel -
15 start_parallel);
```

Results / Output

```
OneDrive\Desktop\problem> cd "c:\Use
serial matrix multiplication: 11236
parallel matrix multiplication: 2482
OneDrive\Desktop\problem>
```

Conclusion

- In conclusion, parallel programming can significantly improve the performance of computationally intensive tasks such as matrix multiplication.
- The OpenMP library provides a simple and effective way to parallelize code, and with careful optimization, even more, considerable speedups can be achieved.