

TA Session 3

조교 김진표, 박대영, 신준식, 정재훈
pp-ta@aces.snu.ac.kr



THUNDER Research Group
Seoul National University
서울대학교 천동 연구실



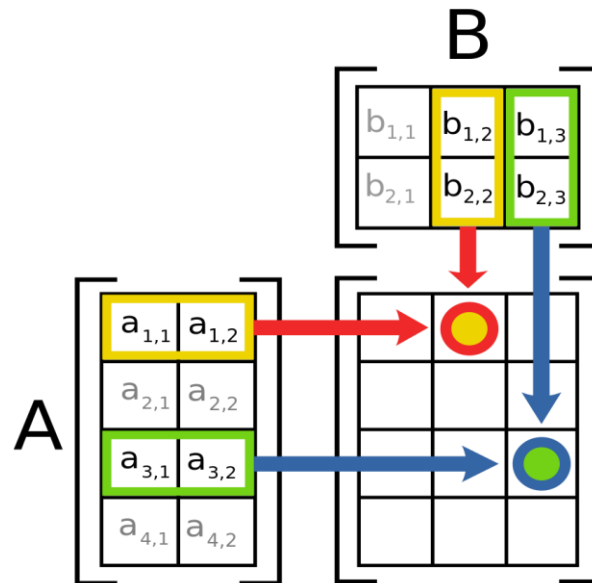
구성

- 과제2 Pthread matmul 해설
- 과제3 OpenMP matmul 소개
- 과제3 진행



과제3: OpenMP matmul

- FP32 행렬 둘을 곱하는 프로그램을 OpenMP 를 이용해 최적화
- 과제3 은 채점하지 않음
- 과제4 는 과제3에 MPI 를 추가해 Multi-node matmul 을 구현할 것임
- 과제4 를 채점함



OpenMP matmul

- /skeleton/HW3 에 주어진 뼈대 코드를 각자의 홈 디렉토리로 복사해 작업
 - `cp -r /skeleton/HW3 ~/`
 - Makefile, run.sh, ...
 - make 로 컴파일이 가능하도록 Makefile 을 제공
 - 실행을 위한 run.sh 스크립트를 제공
- 뼈대 코드를 수정해 행렬 곱셈을 최적화하는 것이 목표
 - mat_mul.cpp 만 수정 가능



OpenMP matmul - 실행 예시

- Pthread matmul 과 같음
- make 로 컴파일
- run.sh, run_performance.sh, run_validate.sh 으로 실행
 - srun 을 통해 계산 노드에서 실행하는 커맨드



OpenMP matmul - 평가 기준

- OpenMP matmul 은 채점하지 않음
- 성능 기준은 Pthread matmul 과 같음



OpenMP matmul - 생각해 볼 것

- CPU의 Theoretical peak FLOPS 및 Pthread matmul 과 달성한 성능 비교
- OpenMP 의 여러 schedule 방식의 비교
 - 어떤 것들이 있는지 (hint: OpenMP 4.5 기준 5개)
 - matmul 에는 어떤 것이 가장 어울릴지
- OpenMP 와 Pthread 사용 비교



OpenMP matmul - 주의사항

- 사용한 스레드 개수를 셀 때, 계산을 수행한 스레드만 개수를 셈
- 반드시 주어진 num_threads 이하의 스레드를 사용하도록 구현해야 함
 - 몰래 더 많은 스레드를 사용해서 성능의 이득을 본 경우 0점 처리
- OpenMP 이외의 병렬화 방식 (Pthread, MPI, OpenCL 등) 사용 금지
 - Vector instruction 은 사용 가능

