

HW#4

1. 주어진 Matrix Multiplication 코드를 수정하여 아래의 사항들을 확인하시오

- A. Lecture 16의 7 page를 참조하여 주어진 코드의 Matrix 접근 패턴을 바꾸고 성능을 확인하시오
 - B. 주어진 코드에 Tiling을 적용하시오. Tile 크기를 바꾸어 가며 성능을 측정하고, 실험 환경의 캐시 크기를 고려하여 측정된 성능 변화 추이를 설명하시오
 - C. 주어진 코드를 Pthread를 사용하여 병렬화하고, Weak scalability와 Strong scalability를 확인하시오
- 과제 수행을 위해 뼈대 코드를 담은 tarball이 주어진다. 이를 untar하고, mat_mul.c의 mat_mul 함수를 수정하여 과제를 수행한다
 - A, B의 경우 Matrix 크기를 원하는 대로 수정할 수 있음. 단, 캐시의 효과를 볼 수 있을 만큼 충분히 큰 Matrix 크기를 사용하여야 함
 - A, B는 코드를 제출 할 필요가 없으며, 실험 내용과 실험 결과 분석을 적은 보고서만 제출할 것
 - C는 코드와 보고서를 모두 제출하여야 함
 - 뼈대 코드 35번째 줄에 주석으로 표시 된 부분 아래쪽의 코드는 시간 측정 및 결과 검증을 위한 코드이므로 수정하지 말 것 (Runtime argument등을 추가로 받기 위해 아래쪽의 코드를 수정한 경우, 보고서에 이를 적을 것)
 - Matrix 크기는 실험 시 원하는 대로 수정할 수 있음
 - C의 경우, 채점의 편의를 위해 제출 코드의 디렉토리에서 **make run**을 수행 시 **4개의 Thread를 사용하여 2048 X 2048 matrix multiplication**을 수행하는 작업을 **천둥 작업 스케줄러에 enqueue하도록** 코드 및 Makefile을 설정하여 제출할 것

2. K-means 알고리즘은 주어진 데이터를 k개의 클러스터로 묶는 알고리즘으로, 대용량의 데이터를 분류하는 데에 유용하게 사용된다. 주어진 k-means 구현을 Pthread를 사용하여 병렬화하고, Weak scalability와 Strong scalability를 확인하시오

- 과제 수행을 위해 뼈대 코드, 데이터 생성 스크립트, 결과 visualization 스크립트를 포함한 tarball이 주어진다. K-means 알고리즘의 동작 원리 및 데이터 생성 스크립트, 결과 Visualization 스크립트는 별첨한 ppt 및 뼈대 코드의 Makefile을 참조할 것
- 뼈대 코드의 kmeans_seq.cpp 파일을 참조하여 **kmeans_pthread.cpp** 파일에 **병렬 kmeans 알고리즘을 구현**할 것
- 병렬화된 K-means 알고리즘의 수행 결과는 병렬화 전의 K-means 알고리즘의 수행 결과와 같아야 함
- Data point 개수, Centroid 개수, Iteration 횟수 등은 실험 시 원하는 대로 수정할 수 있음
- 소스 코드 및 보고서를 제출
- 채점의 편의를 위해 제출 코드의 디렉토리에서 **make run**을 수행 시 **4개의 Thread**를 사용하여 **65535개의 data point**를 **64개의 Centroid**를 사용하여 분류하는 작업을 **1024번의 iteration**을 통해 수행하는 작업을 **천둥 작업 스케줄러에 enqueue**하도록 코드 및 Makefile을 설정하여 제출할 것

과제 기한 : 5월 6일 23시 59분

제출물 : 보고서(1.A, 1.B, 1.C, 2 내용 포함), 1.C번 과제 소스코드 tarball, 2번 과제 소스코드 tarball

제출 방법 : 조교 메일(wookeun@aces.snu.ac.kr)로 제출