

2분반 – 201910899 – 배수한

4.13

풀이과정 :

1 각 스레드는 같은 데이터를 전달받고, 이를 통해 서로 다른 작업을 수행한다.

따라서 이 프로그램은 테스트 병렬성을 보인다.

2 스도쿠 검증기는 각 행, 열, 3x3 소격자에 대한 다른 데이터를 전달받는다.

또한 이에 대해 서로 행, 열, 소격자에 그 숫자가 있는지 확인하는 것에 대한 다른 작업을 수행한다.

따라서 이는 테스트 병렬성, 데이터 병렬성을 보인다.

3 정수 목록을 이등분해서 각 스레드에 들어가고, 사용자가 선택한 정렬 알고리즘을 통해 같은 작업을 수행한다. 각 스레드는 서로 다른 데이터를 전달받으므로 데이터 병렬성을 보인다.

4 다중 스레드 웹서버는 서버로부터 할 작업을 전달받고, 새로운 스레드를 생성해 전달받은 작업을 수행한다. 서버는 추가적인 클라이언트 요청을 listen하는 작업을 재개한다. 따라서 이 웹서버는 테스트 병렬성을 보인다.

정답: 테스트 병렬성/데이터,테스트 병렬성/데이터 병렬성/테스트 병렬성

4.17

풀이과정:

부모프로세스가 자식프로세스를 생성한다. 자식프로세스는 새로운 스레드를 생성해 runner함수를 실행한 후 스레드의 종료를 기다린다. 따라서 자식프로세스에서의 value 값은 5가 된다.

부모프로세스에서의 value값은 변함이 없으므로 값은 0이 된다.

정답: CHILD: value = 5

PARENT: value = 0

5.10

증명:

상호배제 – while(flag[j])를 통해 임계구역에 진입하기 전에 pj가 임계구역에 먼저 진입했는지 확인한다. 따라서 두 프로세스가 동시에 임계구역에 접근하는 것이 불가능하므로 상호배제 조건을 만족한다.

진행(progress) – 나머지구역에 진입하기 전에 flag[i]=false를 통해 다음에 진입할 프로세스의 순서를 결정하는 것에 대해 참여한다. 따라서 progress조건을 만족한다.

한정된 대기(Bounded waiting) - while문을 돌면서 대기중인 j가 실행할 수 있게 turn=j를 통해 다음 순서에 j

가 실행할 수 있게 도와준다. 따라서 Bounded waiting조건을 만족한다.

5.13

- 다중 처리기 환경에서 인터럽트를 사용할 경우 다른 처리기에서 실행되는 상이한 프로세스들의 명령어들이 임의의 방법으로 서로 끼어들 수 있다. 예를 들어 인터럽트 처리 루틴이 공유자원에 접근하는 경우 다른 프로세서와 충돌이 발생할 수 있다. 따라서 다중 처리기 시스템에서는 모든 처리기에서 인터럽트를 금지해야 한다.