

제 1 교시

국어 영역

홀수형

[1~3] 다음 글을 읽고 물음에 답하시오.

현대 사회에 없어서는 안 될 도구인 컴퓨터의 가장 핵심적인 부품은 CPU이다. CPU는 컴퓨터에서 프로그램을 실행하는데 필요한 각종 연산을 수행한다. 이 과정에서 CPU는 메모리에 저장된 명령어나 데이터와 같은 자료들을 가져오고 그를 내부 연산, 즉 해석, 실행한 후 결괏값을 메모리에 저장한다. 그런데 CPU는 메모리에서 값을 읽거나 기록하는 과정에서 내부 연산 대비 수백 배에 달하는 지연을 발생시킨다. 따라서 CPU 상에서 자료를 처리한 후 다음 자료가 입력될 때까지 대기하는 동안 병목 현상이 발생하고, 이는 CPU의 성능 하락으로 이어진다. 이러한 문제점은 CPU가 자주 쓰는 자료를 미리 저장해둘 수 있는 ㉠ 캐시 메모리를 설치함으로써 해결할 수 있다. 그런데 캐시 메모리는 어떤 원리로 CPU의 병목 현상을 해결할 수 있는 걸까? 이를 알기 위해서 캐시 메모리의 역할과 캐시 메모리가 데이터를 어떻게 읽고 쓰는지 알아보자.

컴퓨터는 여러 종류의 메모리를 장착하고 있다. 일반적으로는 레지스터, L1, L2, L3 캐시 메모리, ㉡ RAM를 장착하고 있고, 이들은 나열한 순서대로 계층을 이루고 있으며, 메모리에 접근하는 속도는 빠르지만 용량은 작다. 즉, 레지스터는 가장 높은 계층에서 가장 빠른 접근 속도와 가장 작은 저장 용량을 가지고 있고, RAM은 가장 낮은 계층에서 가장 느린 접근 속도와 가장 큰 저장 용량을 가지고 있다. 여기에서 레지스터는 수 KB에 달하는 용량을 가지지만 캐시 메모리는 보통 수 MB에 달하는 용량을 가지고, RAM은 보통 수 GB에 달하는 용량을 가진다. 그런데 레지스터는 설계 목적 상 명령어보다는 연산에 필요한 데이터를 저장하는 데 주로 쓰이고, RAM은 접근 속도가 과도하게 느리므로 CPU에 자료를 전달할 때 병목 현상이 발생한다. 이런 이유에서 필요한 것이 레지스터와 RAM의 중간 정도의 성능을 가진 캐시 메모리이다.

캐시 메모리는 저장 용량이 보통 32바이트에서 128바이트 사이인 캐시 라인과 대개 8개의 캐시 라인으로 구성된 세트 이루어져 있다. 캐시 라인은 블록이라고도 불리며, 여기에는 RAM에 저장된 자료를 옮겨 저장해둘 수 있다. 이 때 캐시 라인에는 CPU에서 자주 쓰이는 자료만이 아닌 그 자료 주변에 위치한 다른 자료들도 같이 저장된다. 이는 프로그램에서 몇몇 특정 자료들의 사용 빈도가 타 자료에 비해 높고, 사용 빈도가 높은 자료 주변의 자료들도 곧 쓰일 가능성이 높다는 시공간적 지역성 때문이다. 이런 방식으로 캐시 메모리에 자료를 저장해두면 데이터의 시공간적 지역성을 활용해 CPU에 자료를 빠르게 전달할 수 있다.

캐시 메모리에 저장된 자료를 읽기 위해서는 먼저 각 자료의 위치를 알아야 한다. RAM의 각 위치는 고유한 메모리 주소를 가진다. RAM에 존재하던 어떤 자료가 캐시 메모리에 메모리 주소와 함께 입력되면 캐시 메모리는 메모리 주소를 보통 인덱스, 태그, 오프셋으로 나누는데, 이는 각각 세트, 라

인, 라인 내 바이트 위치를 가리키는 데 사용된다. CPU가 특정 메모리 주소에 자료 요청 명령을 보내면 캐시 메모리는 인덱스를 기반으로 세트를 선택하고, 세트 내 태그를 비교해 특정 메모리 주소를 가진 자료가 라인 내에 존재하는지 확인한다. 만약 해당 라인 내에 요청 자료가 존재하면 오프셋 정보를 활용해 해당 정보를 읽고, 정보를 읽는 것이 성공했을 때 이를 ‘히트’라고 부른다. 만약 해당 라인 내에 요청 자료가 존재하지 않으면 그 자료를 찾을 때까지 하위 계층 메모리에 순차적으로 요청을 보낸다. 즉, L1 캐시 메모리에 특정 자료가 없으면 L2 캐시 메모리에 추가 요청을 보내고, L2 캐시 메모리에도 특정 자료가 없으면 L3 캐시 메모리에 추가 요청을 보낸다. 이렇게 원하는 자료가 캐시에 없을 때를 ‘미스’라고 부른다.

캐시 메모리에 정보를 쓸 때도 마찬가지로 특정 메모리 주소의 캐시 메모리 내 존재 여부를 따져야 한다. 이 때 캐시 메모리 내에 정보를 쓰려는 특정 주소가 존재할 때를 ‘히트’, 특정 주소가 존재하지 않을 때를 ‘미스’라고 부른다. 만약 캐시 메모리가 ‘히트’ 상태이면 CPU는 캐시 메모리의 쓰기 정책에 따라 라이트스루 혹은 라이트백 방식으로 캐시 메모리에 값을 쓴다. 전자는 값을 쓸 때 캐시 메모리에 그 값을 기록한 직후 하위 계층의 메모리에도 동일한 값을 전송하는 방식이고, 후자는 캐시 메모리에 그 값을 기록한 후 값을 기록한 라인의 더티 비트 값만 0에서 1로 바꾸고, 차후에 캐시 메모리에 기록한 값이 RAM에 전송되면 더티 비트 값을 다시 0으로 바꾸는 방식이다. 여기서 더티 비트는 캐시 라인과 RAM의 특정 메모리 주소의 값이 변경되었는지를 나타내는 일종의 표시용 비트이다. 반대로 캐시 메모리가 ‘미스’ 상태이면 라이트 얼로케이트 혹은 노 라이트 얼로케이트 방식으로 캐시 메모리에 값을 쓴다. 전자는 쓰려는 특정 주소를 하위 계층 메모리에서 불러와 캐시 메모리에 입력한 후 캐시 메모리에 값을 쓰는 방식이고, 후자는 쓰려는 특정 주소를 캐시 메모리에 불러오지 않고 바로 하위 계층 메모리에 쓰는 방식이다.

8. 윗글에서 알 수 있는 내용으로 옳은 것은?
- ① CPU가 어떤 값을 읽는 속도는 연산을 수행하는 속도보다 빠르다.
 - ② CPU는 쓰기 과정에서 오직 캐시 메모리에만 값을 쓸 수 있다.
 - ③ CPU의 레지스터에는 연산에 필요한 데이터만이 저장될 수 있다.
 - ④ 캐시 메모리에는 CPU가 자주 사용하는 정보만이 저장될 수 있다.
 - ⑤ 캐시 메모리는 ‘미스’ 상태일 때 RAM에서 자료를 찾아올 수 있다.

9. ㉠과 ㉡을 비교한 내용으로 옳은 것은?

- ① ㉠과 ㉡은 모두 메모리 주소가 고정적이다.
- ② ㉠과 ㉡에서 모두 데이터를 불러올 수 있다.
- ③ ㉠은 ㉡과 달리 저장 용량이 MB 단위를 넘어갈 수 없다.
- ④ ㉡은 ㉠과 달리 메모리의 상태를 나타내는 비트를 가진다.
- ⑤ ㉡은 ㉠과 달리 CPU가 정보를 쓸 때 항상 그 정보를 입력 받는다.

10. <보기>는 캐시 메모리가 L1, L2, 캐시 메모리로 구성된 컴퓨터가 작동하는 과정이다. <보기>에 관한 설명으로 옳지 않은 것은? [3점]

<보 기>

컴퓨터는 메모리 주소가 0x2001인 자료 a와 주소가 0x3004인 자료 b를 불러온 후, 두 자료를 가지고 연산한 값을 주소가 0x5000인 저장 공간에 저장했다. 이 때 메모리 주소가 0x3으로 시작하는 자료는 CPU에서 자주 사용하는 자료이고, 메모리 주소가 0x2001인 자료는 캐시 메모리에 저장되어 있지 않다.

- ① 컴퓨터가 a를 불러올 때 걸린 시간이 b를 불러올 때 걸린 시간보다 더 길겠군.
- ② 컴퓨터가 연산한 값을 저장할 때 RAM에는 그 값이 저장되지 않을 수 있겠군.
- ③ 컴퓨터가 연산한 값을 저장할 때 캐시 메모리에 그 값이 저장되지 않을 수 있겠군.
- ④ 만약 L2 캐시 메모리에 a가 없다면 컴퓨터는 a를 RAM에서 불러오겠군.
- ⑤ 컴퓨터는 a를 불러올 때 주변의 다른 자료도 같이 불러오겠군.

* 확인 사항

- 답안지의 해당란에 필요한 내용을 정확히 기입(표기)했는지 확인 하시오.
- 이어서, 「선택과목(화법과 작문)」 문제가 제시되오니, 자신이 선택한 과목인지 확인 하시오.