

```
created: 2022-12-01
last-modified: "2022-12-06"
tags: study, university
```

31	6	5	4	3	2	1	0
Tag				Index		Word Offset	

	V	Tag	Data1	Data2	Data3	Data4	Dirty
#0	1b	26b	32b	32b	32b	32b	1b
#1							
#2							
#3							

캐시 진행

sort는 총 0x20번, 즉 32번 실행된다.

첫 루프

첫 루프의 진행은 아래와 같다.

Input Data 읽기

Input Data가 0x400에서 시작하므로 청크의 시작점이다.

- lw s0, 0(a0)
 - 첫 4개의 Input Data를 불러옴
 - #0에 사상
- lw s2, 0(a0)
 - 1번과 같은 데이터를 불러옴
 - 캐시 히트!
- lw s0, 0(a0)
 - 전체 횟수의 $\frac{1}{4}$ 은 캐시 미스
 - 전체 횟수의 $\frac{3}{4}$ 은 캐시 히트
- lw s2, 0(a0)
 - 항상 캐시 히트

위 과정을 32번 반복한다. 이때, a0는 4씩 증가되므로 모든 32개의 데이터를 순서대로 읽게 된다. 이때 발생하는 캐시 히트와 캐시 미스 횟수를 구해보면 다음과 같다.

이때, 한 루프(32번의 읽기) 이후 캐시 상태를 보자면, Input Data의 뒤쪽 절반(16개) 데이터가 순서대로 캐시에 차 있을 것이다.

Output Data 쓰기

이후, 다음 두 sw 명령어가 실행된다. (이때, Output Data는 0x480에서 시작하므로 청크의 시작점이다.)

1. sw s0, 0(a1)

- 첫 4개의 Output Data를 불러옴
- #0에 사상

2. sw t0, 0(a4)

- a4는 현재 입력에서 가장 큰 수의 위치
- 첫 루프에서, 그 위치는 #2에 사상되는 15

이 이후 캐시 상태를 보자면, Output Data가 들어 있는 엔트리가 하나 존재하고, 현재 Input Data에서 가장 큰 수의 위치가 들어 있는 엔트리가 하나 존재하고(이 두 엔트리가 같은 엔트리면 후자로 덮어쓰워진다), 나머지 엔트리는 Input Data의 뒤쪽 절반 데이터가 순서대로 캐시에 들어 있다.

두 번째부터의 루프

Input Data 읽기

동일한 과정이 반복된다. 이때, 이전 과정에서 sw t0, 0(a4)를 통해 해당 청크를 이미 사소한 경우 캐시 히트가 발생한다. 캐시가 16개의 데이터를 담을 수 있으므로 원래 캐시 미스인 16번은 캐시 히트가 된다. 각 과정에서 항상 마지막에는 뒤쪽 절반의 데이터가 캐시에 들어 있다.

Output Data 쓰기

sw s0, 0(a1)는 항상 캐시 미스인데, 이전 상황에 캐시에 들어있는 데이터는 Input Data임이 보장되기 때문이다. sw t0, 0(a4)는 16번의 뒤쪽 절반 데이터가 상황에 따라 히트하거나 미스하는데, 해당 데이터가 있는 엔트리가 정렬 후 뒤쪽 절반에 오는 데이터(음수)가 아니거나 이전 명령어에 의해 Output Data로 대체되었다면 캐시 미스, 그렇지 않다면 캐시 히트가 된다. 직접 경우의 수를 확인하면 13번 히트하고 3번 미스한다는 것을 확인할 수 있다.

전체 진행

메모리 로드, 스토어 명령은 다음의 횟수로 발생한다.

$$32 \times (32 + 32 + 1 + 1) = 2048 + 64 = 2112$$

그 중 실제 메모리가 스토어되는 횟수는 다음과 같으며, 메모리 블록의 수는 Input Data 8개, Output Data 8개로 총 16개이다.

$$32 \times (1 + 1) - 2 = 62$$

캐시 히트는 다음의 횟수로 발생한다.

$$32 \times (24 + 32) + 16 + 13 = 1792 + 24 = 1821$$

캐시 히트 레이트는 다음과 같다.

$$\frac{1821}{2112} \sim 0.86$$

캐시의 마지막 상태는 다음과 같다. sw를 시행한 마지막 두 엔트리는 아직 Dirty임에 유의해야 한다. 이 엔트리들은 다음에 덮어쓰워질 때 값을 메모리에 저장한다.

	V	Tag	Data1	Data2	Data3	Data4	Dirty
#0	1	0x11	-17	-17	-17	-17	0
#1	1	0x11	-17	-17	-17	-17	0
#2	1	0x10	-17	-16	-17	-17	1
#3	1	0x13	-13	-14	-15	-16	1