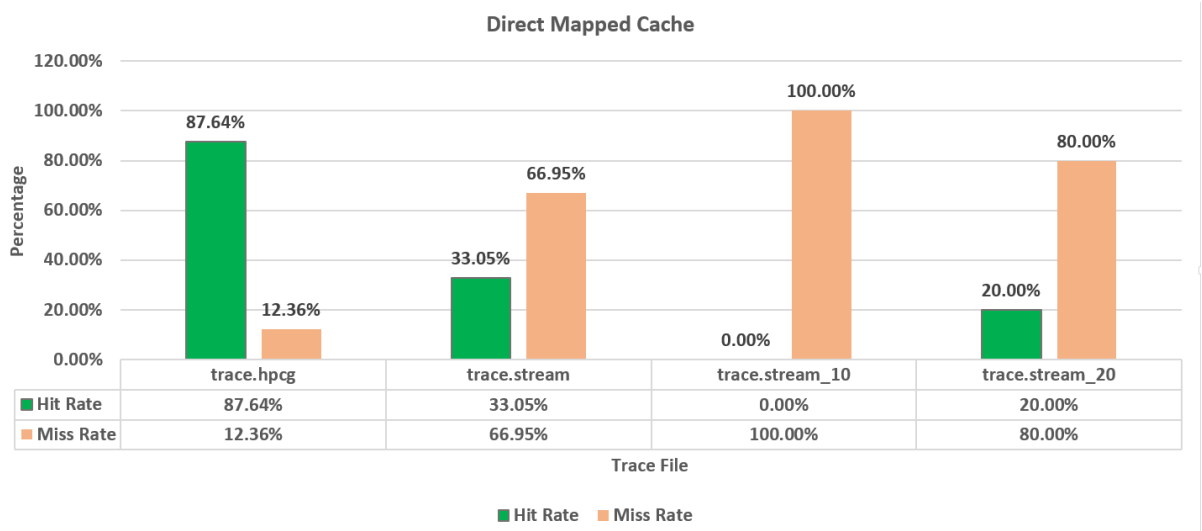
Programming Project #1

Anvesh Muppeda | R11840667

Contents

Part-1 Direct Mapped Cache

Hit Rate and Miss rate for all the trace files respectively by using the Direct Mapped Cache technique.



Part 2 Fully associative and n-way set associative cache

Here I Developed code that will give multiple options to users to switch between the required mapping technique same as below.

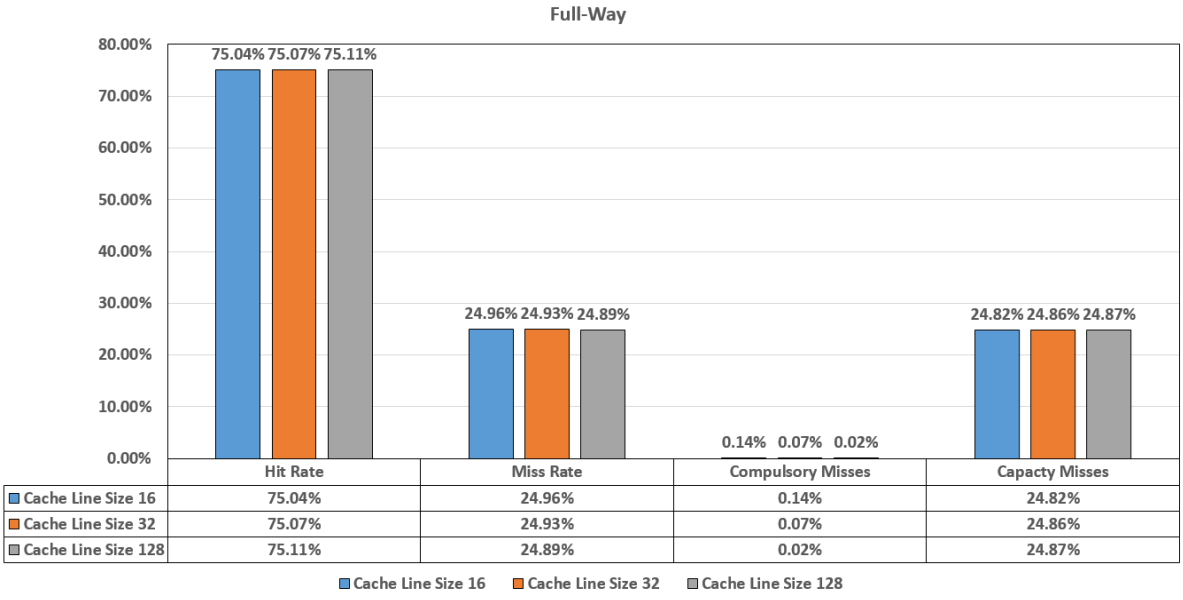
```
login-20-26:/anvesh$
login-20-26:/anvesh$ cd cache
login-20-26:/anvesh/cache$ gcc cachesim.c
login-20-26:/anvesh/cache$
login-20-26:/anvesh/cache$ ./a.out trace_for_students/trace.stream
Enter the one of the Mapping Technique from below list
 1. Direct Mapping
 2. Full Way Mapping
 3. N-Way Mapping
 4. Two-Level Cache(L1 & L2)
 5.Exit
```

Here the user needs to select the required mapping technique to get the results (Hits and Misses).

Part 2.1 Fixed Cache Size

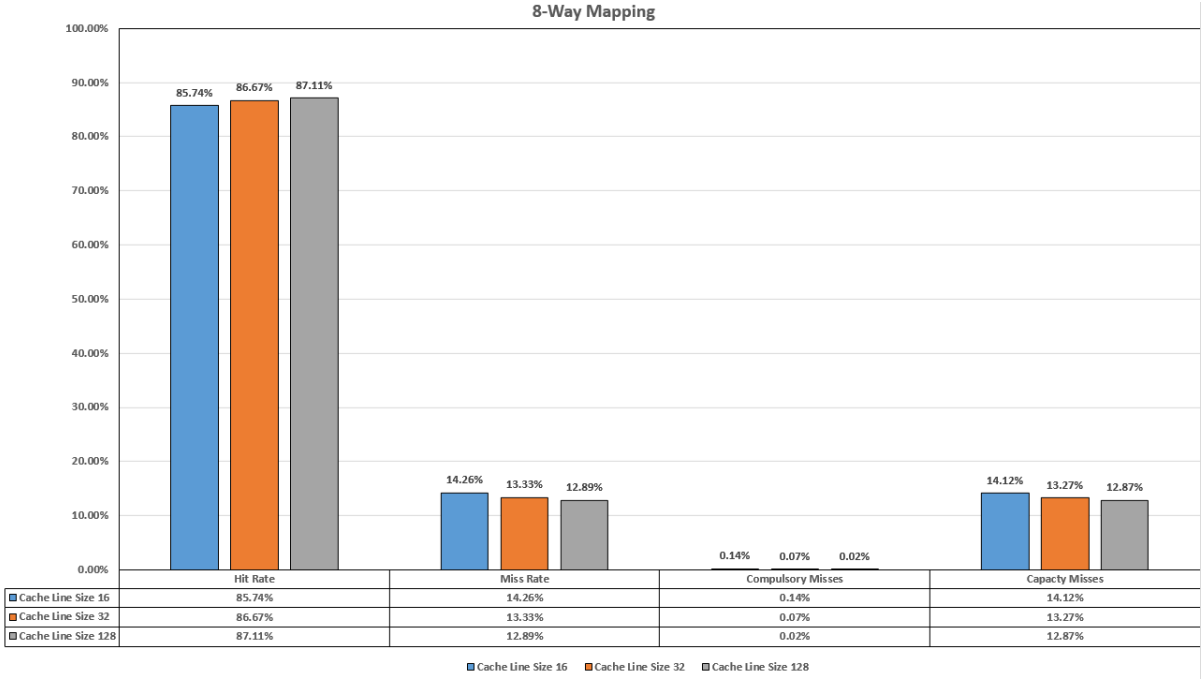
Here cache size is fixed i.e. 32KB. If we increase the cache line size the hit rate needs to be increased. Also, the compulsory misses will decrease which means the miss rate will decrease automatically. The same results can observe in below mapping techniques.

In the below sFull-Way mapping technique, the cache size is fixed (32KB) and by increasing of cache line size to 16B, 32B, and 128B the hit rate also increased to 75.04% 75.07%, and 75.11% respectively. Also, we can observe that compulsory misses will decrease to 0.14%, 0.07%, and 0.02% respectively.

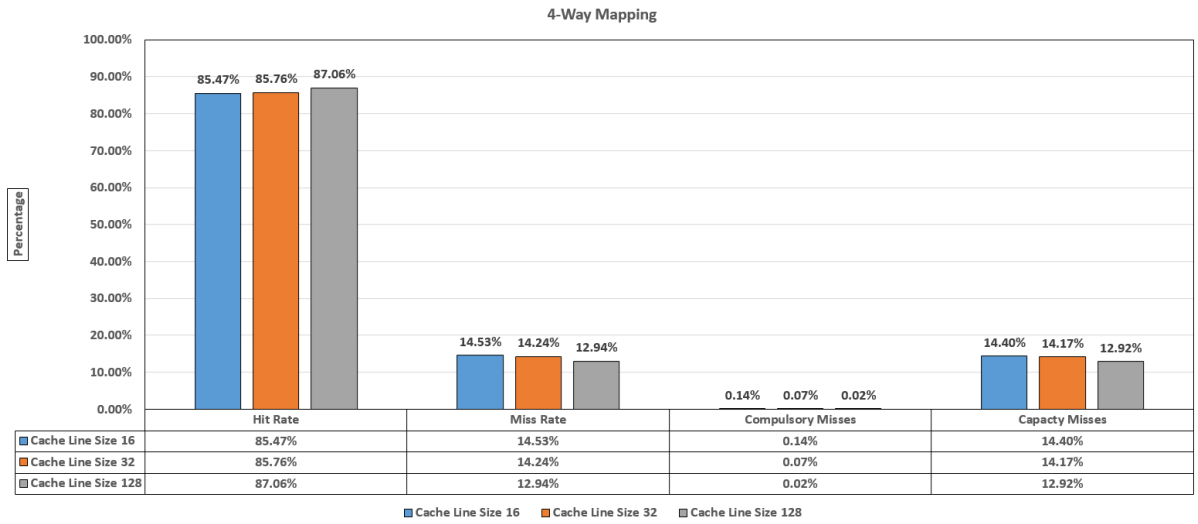


	Hit Rate	Miss Rate	Compulsory Misses	Capacty Misses
Cache Line Size 16	75.04%	24.96%	0.14%	24.82%
Cache Line Size 32	75.07%	24.93%	0.07%	24.86%
Cache Line Size 128	75.11%	24.89%	0.02%	24.87%

In the below 8-Way mapping technique, the cache size is fixed (32KB) and by increasing of cache line size to 16B, 32B, and 128B the hit rate also increased to 85.74%, 86.67%, and 87.11% respectively. Also, we can observe that compulsory misses will decrease to 0.14%, 0.07%, and 0.02% respectively.

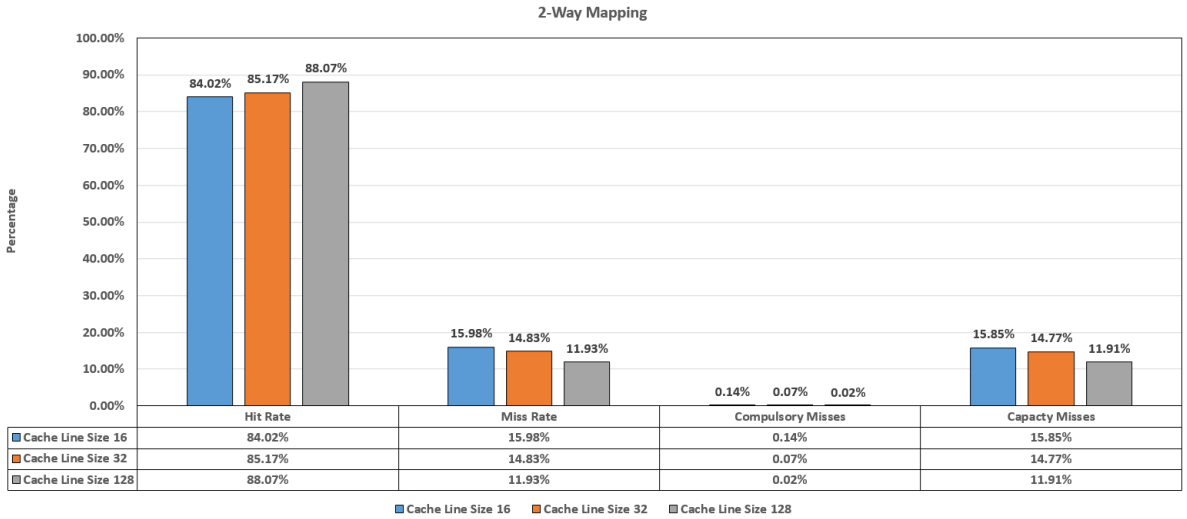


In the below 4-Way mapping technique, the cache size is fixed (32KB) and by increasing of cache line size to 16B, 32B, and 128B the hit rate also increased to 85.47%, 85.76%, and 87.06% respectively. Also, we can observe that compulsory misses will decrease to 0.14%, 0.07%, and 0.02% respectively.



In the below 2-Way mapping technique, the cache size is fixed (32KB) and by increasing of cache line size to 16B, 32B, and 128B the hit rate also increased to 84.02%, 85.17%, and 88.07%

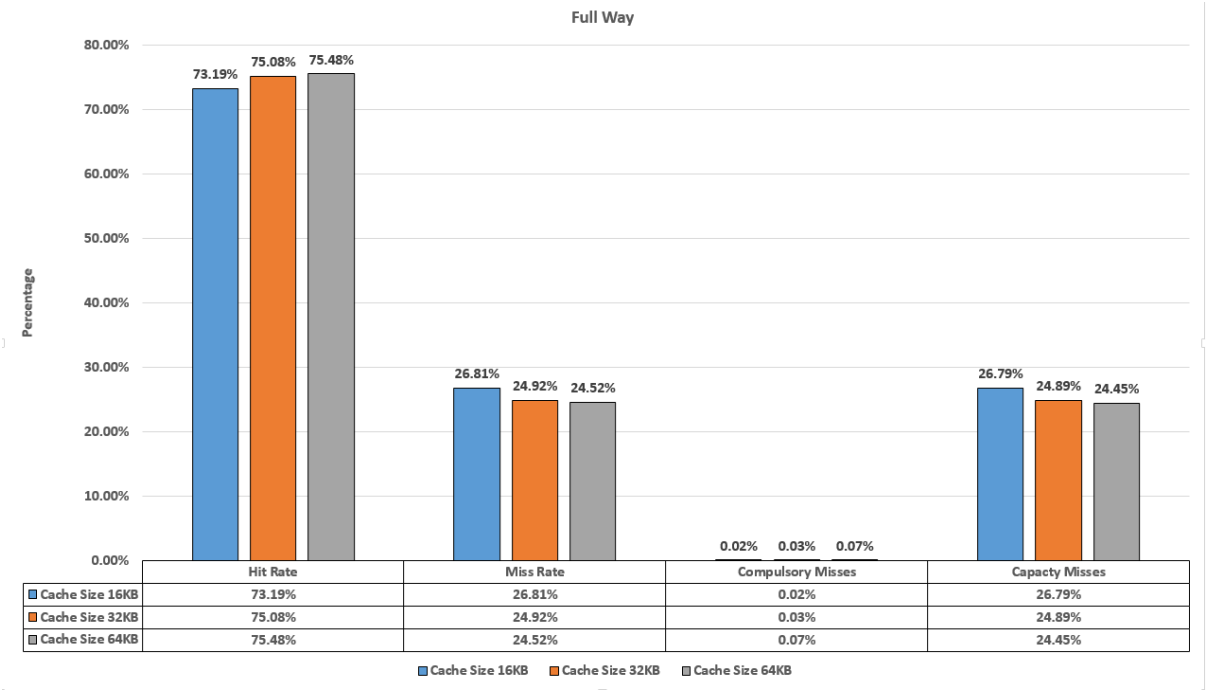
respectively. Also, we can observe that compulsory misses will decrease to 0.14%, 0.07%, and 0.02% respectively.



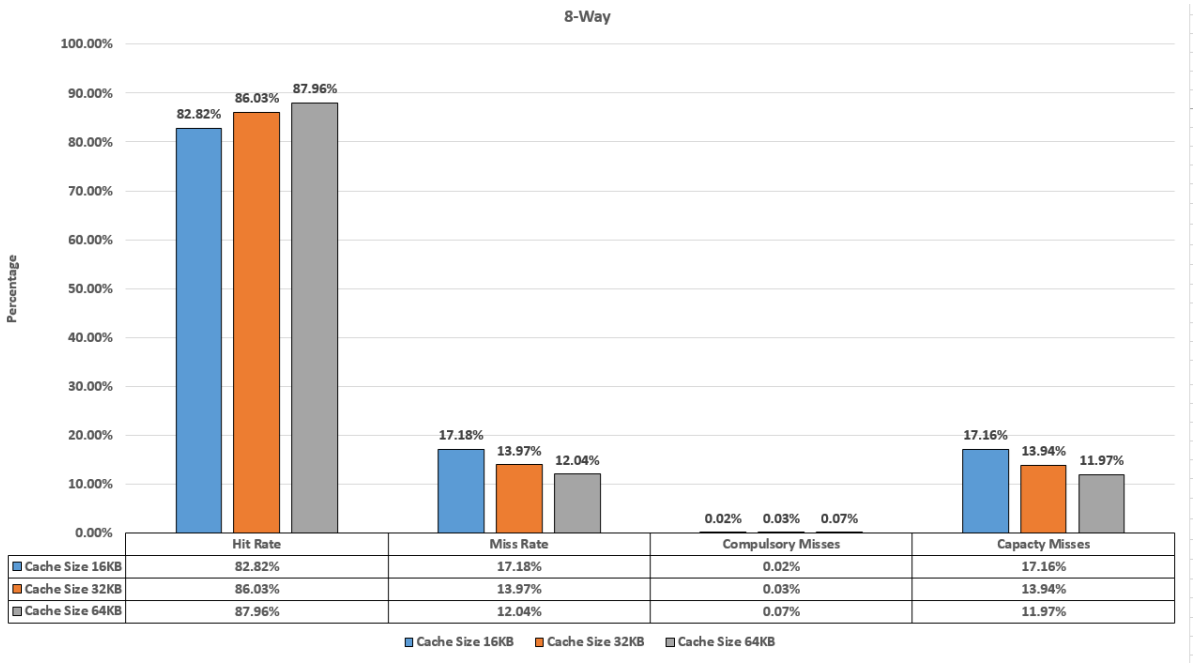
Part 2.2 Fixed Cache Line Size.

In this section, we are given with fixed cache line size i.e. 64B, and by varying the cache size i.e. 16KB, 32KB, and 64KB. We are testing all the different mapping techniques. If we increase the cache size the compulsory misses will increase definitely and the hit rate will increase since we have enough space to store the data.

In the Below Full-Way mapping technique we are increasing the cache size to 16KB, 32KB, and 64KB by maintaining the constant cache line size i.e. 64B, the hit rate will increase to 73.19%, 75.08%, and 75.48% respectively, and also compulsory misses will increase to 0.02%, 0.03%, and 0.07 respectively since the cache size is increased.

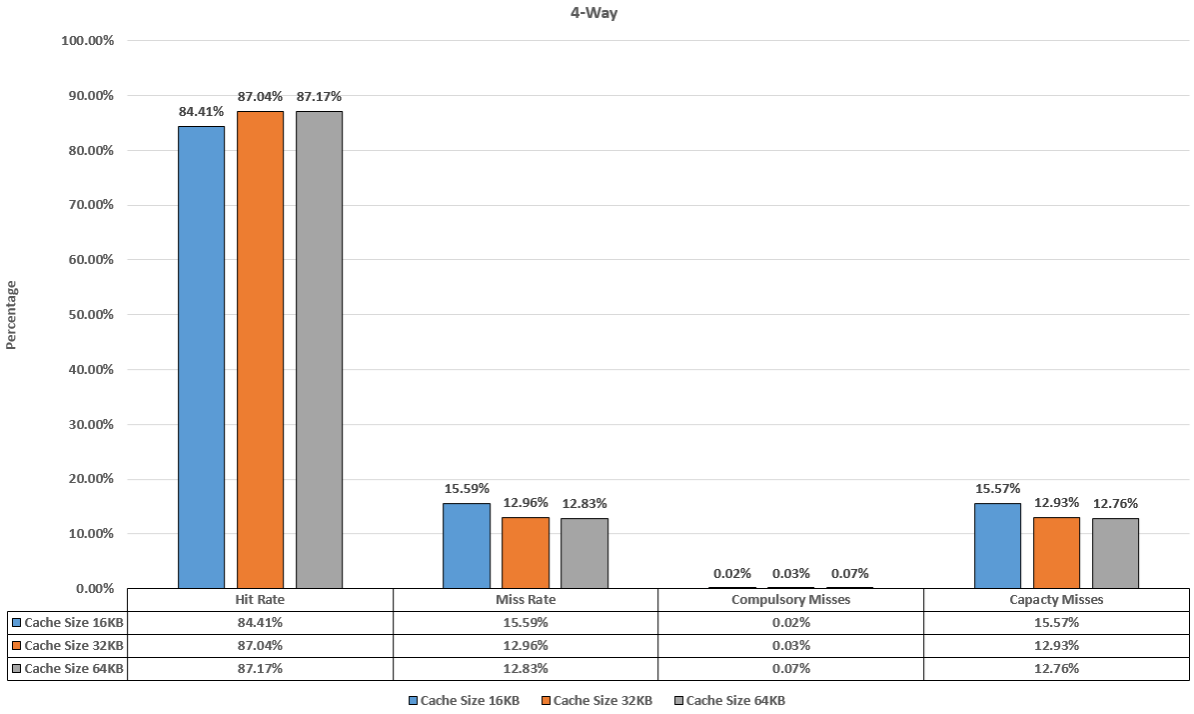


In the Below 8-Way mapping technique we are increasing the cache size to 16KB, 32KB, and 64KB by maintaining the constant cache line size i.e. 64B, the hit rate will increase to 82.82%, 86.03%, and 87.96% respectively, and also compulsory misses will increase to 0.02%, 0.03%, and 0.07 respectively since the cache size is increased.



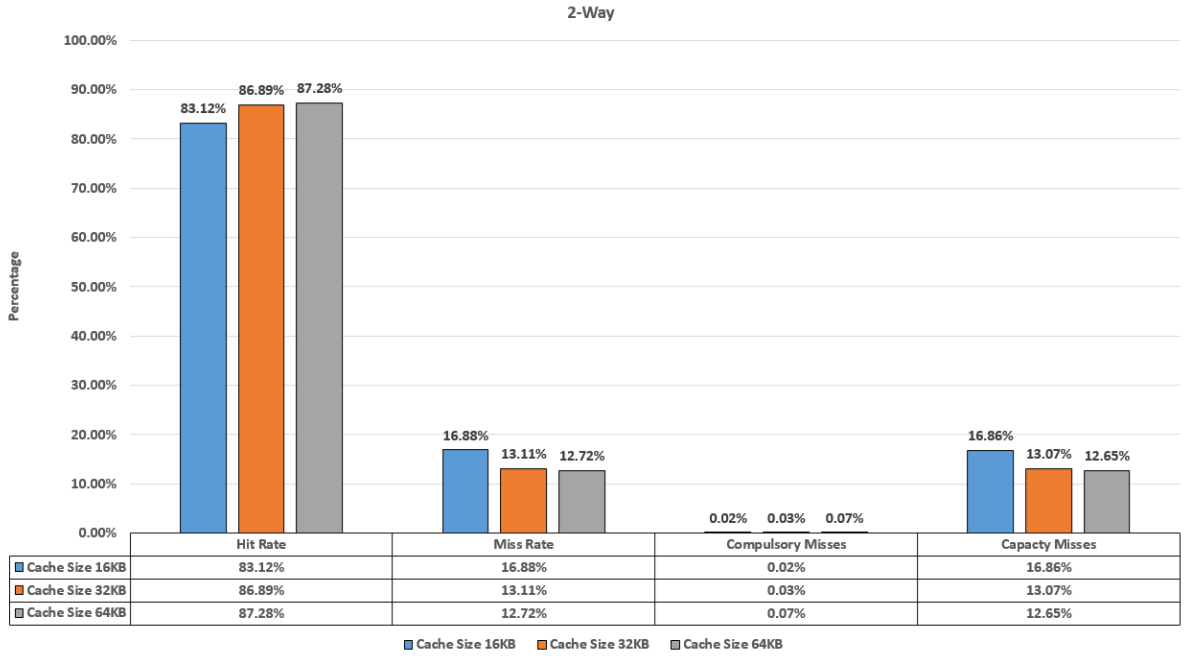
	Hit Rate	Miss Rate	Compulsory Misses	Capacty Misses
Cache Size 16KB	82.82%	17.18%	0.02%	17.16%
Cache Size 32KB	86.03%	13.97%	0.03%	13.94%
Cache Size 64KB	87.96%	12.04%	0.07%	11.97%

In the Below 4-Way mapping technique we are increasing the cache size to 16KB, 32KB, and 64KB by maintaining the constant cache line size i.e. 64B, the hit rate will increase to 84.41%, 87.04%, and 87.17% respectively, and also compulsory misses will increase to 0.02%, 0.03%, and 0.07 respectively since the cache size is increased.



	Hit Rate	Miss Rate	Compulsory Misses	Capacty Misses
Cache Size 16KB	84.41%	15.59%	0.02%	15.57%
Cache Size 32KB	87.04%	12.96%	0.03%	12.93%
Cache Size 64KB	87.17%	12.83%	0.07%	12.76%

In the Below 2-Way mapping technique we are increasing the cache size to 16KB, 32KB, and 64KB by maintaining the constant cache line size i.e. 64B, the hit rate will increase to 83.12%, 86.89%, and 87.28% respectively, and also compulsory misses will increase to 0.02%, 0.03%, and 0.07 respectively since the cache size is increased.



	Hit Rate	Miss Rate	Compulsory Misses	Capacty Misses
Cache Size 16KB	83.12%	16.88%	0.02%	16.86%
Cache Size 32KB	86.89%	13.11%	0.03%	13.07%
Cache Size 64KB	87.28%	12.72%	0.07%	12.65%

Part 3: Two-Level Cache.

In this section, we are given two-level cache i.e. L1 and L2 cache levels.

L1 cache with 64KB cache size, 64B block size, and 2-way mapping technique.

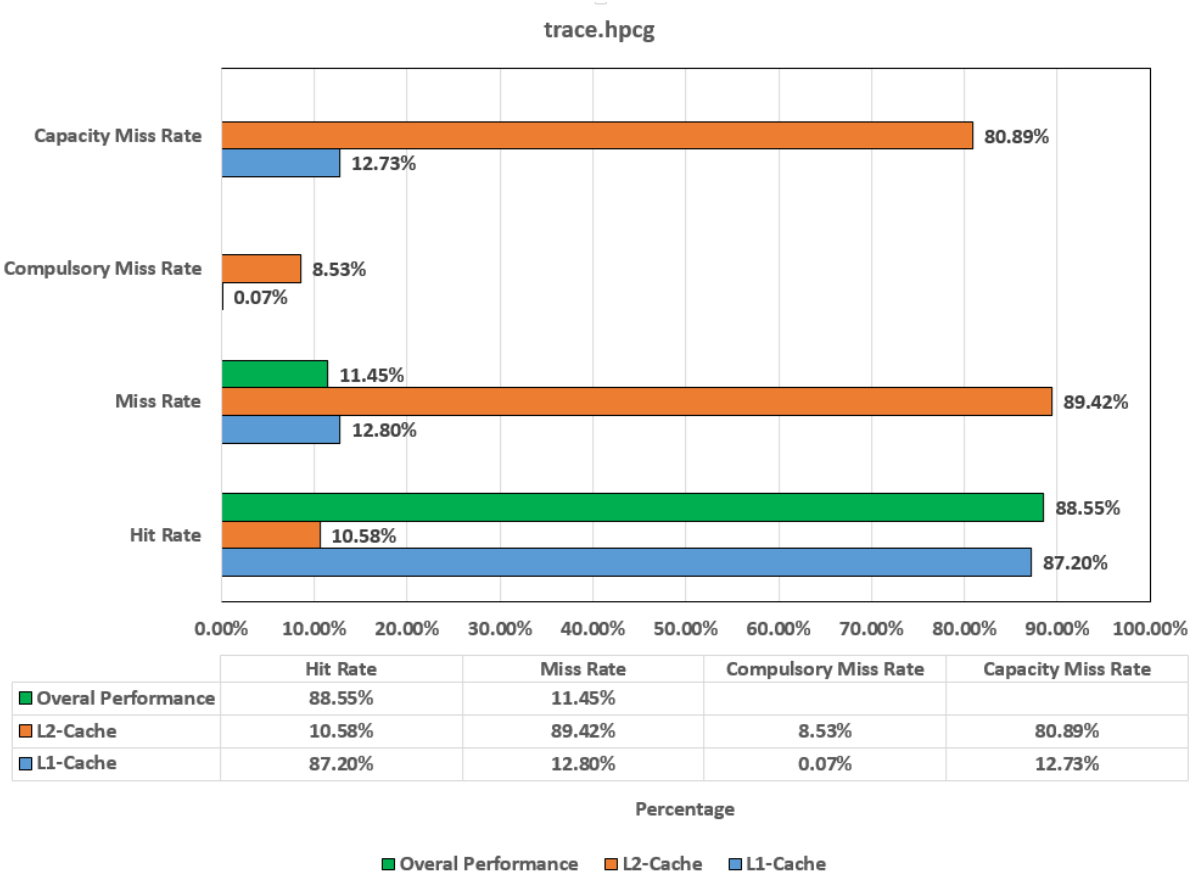
L2 cache with 1MB cache size, 64B block size, and 8-way mapping technique.

Here if anything is missed from an L1 cache then it accesses the L2 for the data, if the data is not present on L2 also, then it will access the data from the main memory. And in my code I am calculating the below values, if the data is present on the L1 cache then it is L1 hit, if the data is not present on the L1 cache then it will hit the L2 cache, if the data is present on L2 cache then it is L2 hit, if it is missed from L2 cache then it is L2 miss.

And the overall performance is calculated as the total number of hits in L1 and L2 are combined as overall cache hits and L2 misses are calculated as overall cache misses.

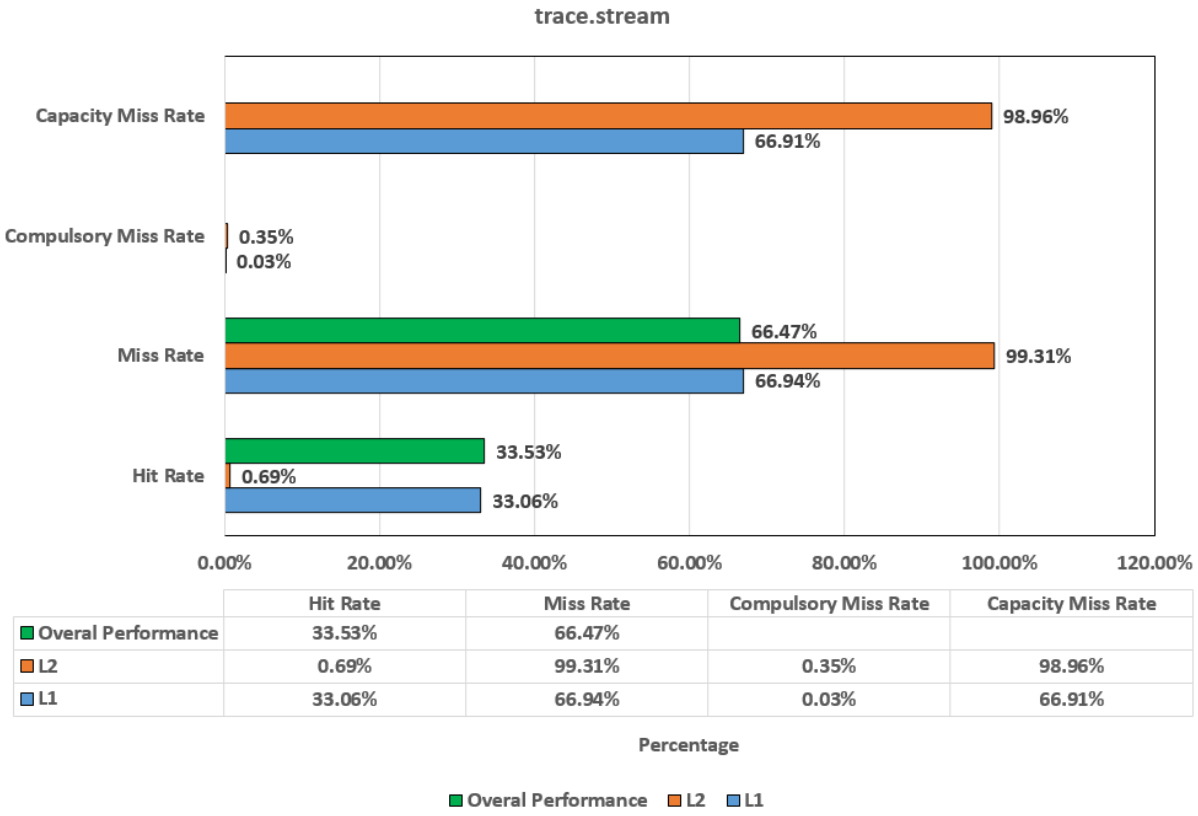
Note: Here I am calculating the L1 hit ratio and miss ratio based on the total access of L1 and calculating the L2 hit ratio and miss ratio based on the total access of L2 which means L1 misses.

The below example is for trace.hpcg trace file, here the overall performance is increased which means the hit ratio is increased to 88.55% and the miss rate is decreased to 11.45%. And hereby L2 cache, we improved the hit ratio.



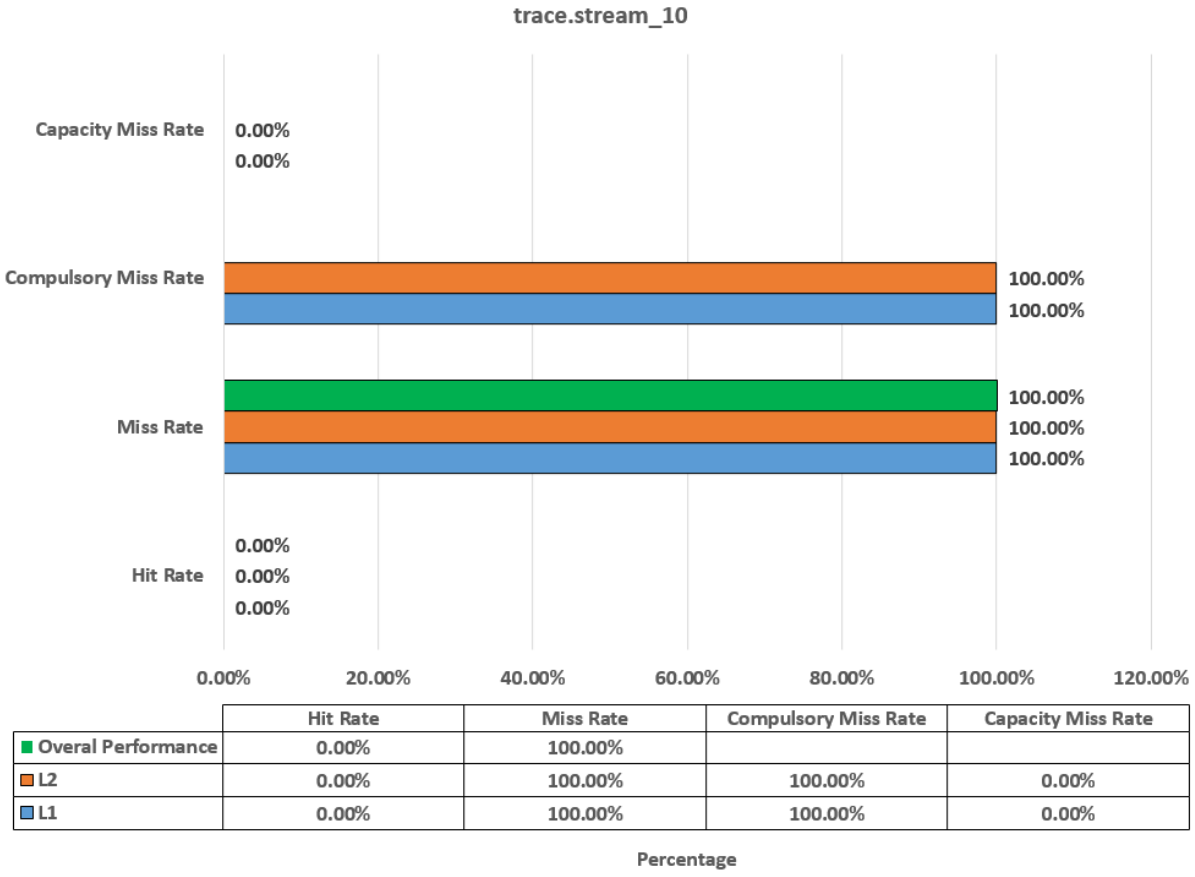
	Hit Rate	Miss Rate	Compulsory Miss Rate	Capacity Miss Rate
Overal Performance	88.55%	11.45%		
L2-Cache	10.58%	89.42%	8.53%	80.89%
L1-Cache	87.20%	12.80%	0.07%	12.73%

The below example is for trace.stream trace file, here the overall performance is increased which means the hit ratio is increased to 33.53% and the miss rate is decreased to 66.47%. And hereby L2 cache, we improved the hit ratio.



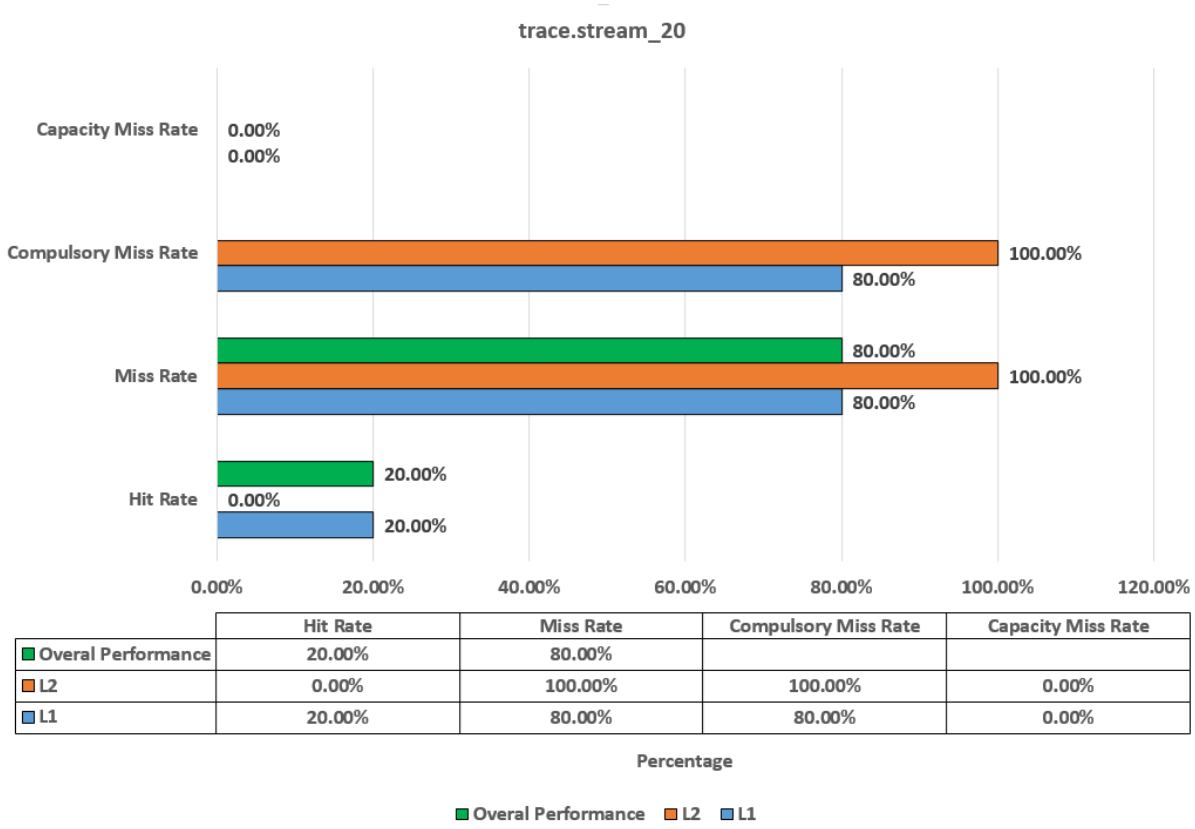
	Hit Rate	Miss Rate	Compulsory Miss Rate	Capacity Miss Rate
Overal Performance	33.53%	66.47%		
L2	0.69%	99.31%	0.35%	98.96%
L1	33.06%	66.94%	0.03%	66.91%

The below example is for trace.stream_10 trace file, here the overall performance is not changed due to the trace file arrangement. In this trace file we are getting all the misses hence we are not able to observe the overall performance.



	Hit Rate	Miss Rate	Compulsory Miss Rate	Capacity Miss Rate
Overal Performance	0.00%	100.00%		
L2	0.00%	100.00%	100.00%	0.00%
L1	0.00%	100.00%	100.00%	0.00%

The below example is for trace.stream_20 trace file, here the overall performance is not changed due to the trace file design, all the L1 misses are missed in L2 as well, hence we didn't see the change in overall performance w.r.t. the L1 performance.



	Hit Rate	Miss Rate	Compulsory Miss Rate	Capacity Miss Rate
Overal Performance	20.00%	80.00%		
L2	0.00%	100.00%	100.00%	0.00%
L1	20.00%	80.00%	80.00%	0.00%