Introduction

For my programming assignment I created a cache simulator that would test different parameters to showcase the effectiveness of different designs, sizes, and replacement methods to get the best hit rates. This project was assigned because it is important to learn about cache memory and all of these aspects that are related to it. Cache memory itself is also important as it is necessary for any computer to work at a reasonable speed, since cache memory helps speed up the processor executions while still keeping its operations at a reasonable cost.

Description of Tests

There are many parameters for each test. To start with the type of cache designs can be chosen between a direct mapped, n way associative, and fully associative design. Then if the n way associative or the fully associative design is chosen then a choice has to be made between the LRU(least recently used) method and the FIFO(first in first out) method to replace lines when needed. Then the cache size is needed in bytes, the size of block in bytes is needed, and with the n way associative design the amount of blocks in a set is needed. After that the program will work with any trace files provided.

Table 1: Byte Size is 64 bytes and trace file used is gcc.trace

Cache Size	Direct Mapped	LRU Fully Associated	FIFO Fully Associated	LRU 2 Way	FIFO 2 WAY	LRU 4 Way	FIFO 4 WAY	LRU 8 Way	FIFO 8 WAY
512	0.785403	0.850598	0.820252	0.828616	0.819662	0.845578	0.826857	0.850598	0.820252
1024	0.831627	0.92603	0.914244	0.891342	0.884788	0.92069	0.909824	0.925231	0.910951
2048	0.889568	0.962729	0.95555	0.921929	0.91493	0.952676	0.945048	0.962731	0.954943
4096	0.948028	0.983416	0.978262	0.967918	0.963724	0.973323	0.966691	0.98178	0.976895
8192	0.963735	0.989703	0.986909	0.98394	0.982295	0.988014	0.985592	0.988941	0.986156
16384	0.981923	0.991404	0.990085	0.989222	0.988266	0.990876	0.989534	0.991264	0.989895

I chose these parameters as the byte size of 64 as it is the common size used for caches and I chose gcc.trace since it is the first trace file. The cache sizes I chose are from the video on the cache analysis project which started with 512 bytes and a 64 byte size. From there each cache size is doubled. The program can alternate between trace files if needed. The direct map has only one column as the only thing to test is the cache size whereas with the fully associative design, it can be tested between the LRU and FIFO design as well. With the n way designs it has 6 columns because it can be tested between LRU and FIFO and at the same time it can also be tested with the n amount.

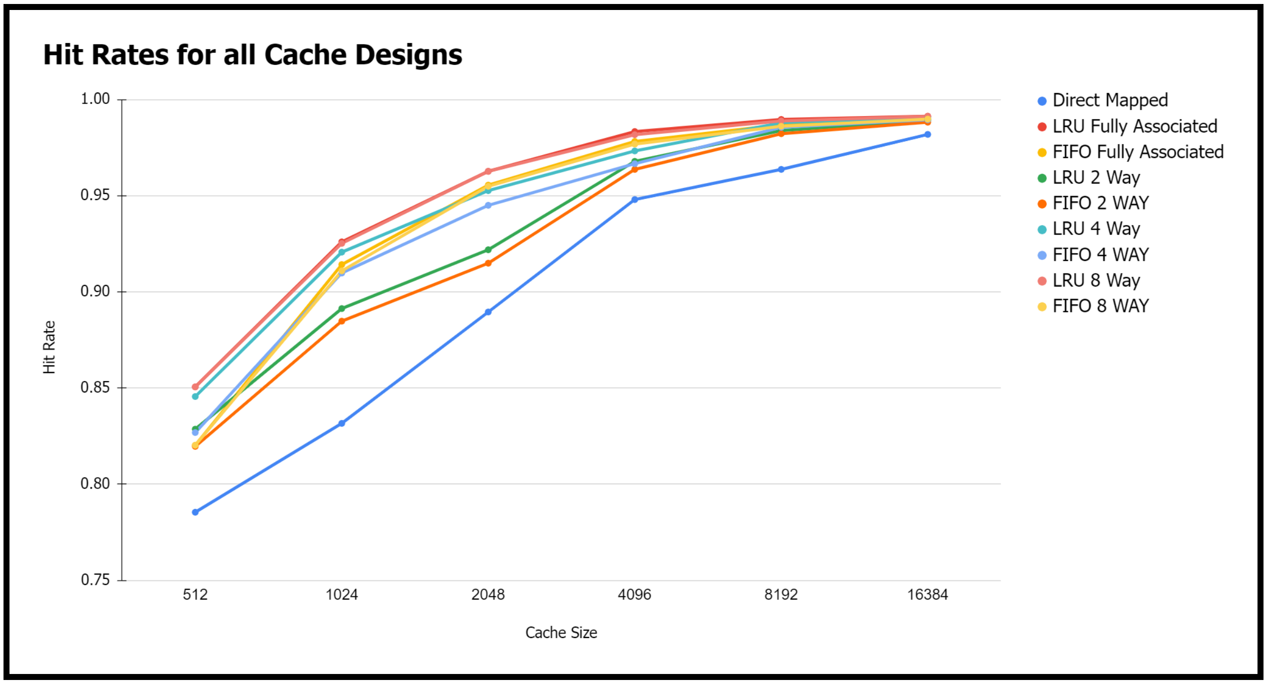
Results

Link to spreadsheet:
https://docs.google.com/spreadsheets/d/1u8N4Or-cXw4ImOxhwOBPa1FVAduvwmIUXBL13BA7LrQ/edit?usp=sharing

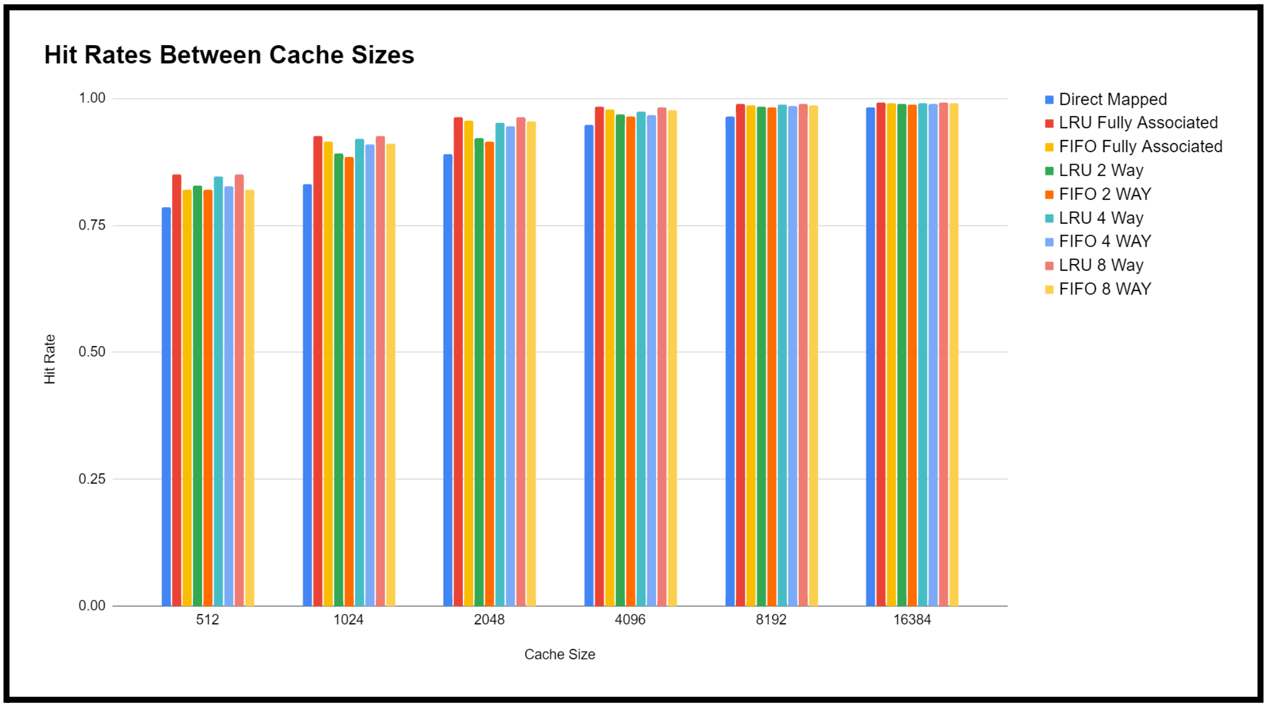
Table 2: For value hit rates.

Cache Size	Direct Mapped	LRU Fully Associated	FIFO Fully Associated	LRU 2 Way	FIFO 2 WAY	LRU 4 Way	FIFO 4 WAY	LRU 8 Way	FIFO 8 WAY
512	0.785403	0.850598	0.820252	0.828616	0.819662	0.845578	0.826857	0.850598	0.820252
1024	0.831627	0.92603	0.914244	0.891342	0.884788	0.92069	0.909824	0.925231	0.910951
2048	0.889568	0.962729	0.95555	0.921929	0.91493	0.952676	0.945048	0.962731	0.954943
4096	0.948028	0.983416	0.978262	0.967918	0.963724	0.973323	0.966691	0.98178	0.976895
8192	0.963735	0.989703	0.986909	0.98394	0.982295	0.988014	0.985592	0.988941	0.986156
16384	0.981923	0.991404	0.990085	0.989222	0.988266	0.990876	0.989534	0.991264	0.989895

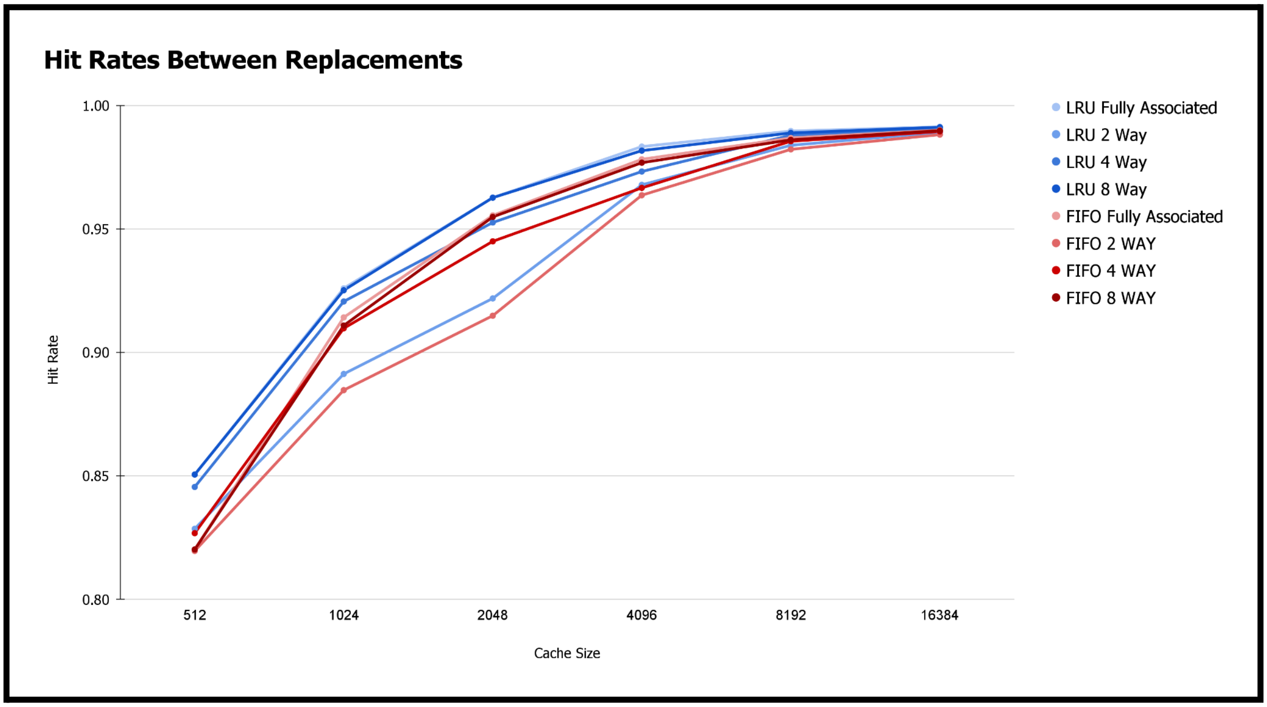
Graph 1: Hit Rate Comparisons Between All Cache Designs



Graph 2: Hit Rates Between Cache Sizes, Higher is Better



Graph 3: Hit Rate Comparisons Between Replacements, LRU are blue shades, FIFO are red shades.



Conclusions

From the results gathered from my cache simulator I can say that direct mapped caches are the worst option as a cache as it has the lowest hit rate at every cache size. From there the fully associative and the n way associative designs are mixed together. The best cache design in terms of hit rate is the fully associative LRU design with the 8 way LRU design being a very close second. The two specific designs start at the same hit rate with the 512 cache size but as the cache size grows, the fully associative designs have a slightly more increased hit rate compared to the 8 way design.

In terms of replacement methods between the LRU and FIFO, the LRU designs outperformed each FIFO design for both design types, being the fully associative and n way associative. The larger n way designs all have better hit rates than the smaller n way designs. This can be easily seen in graph 3 where all of the n way designs are on the line graph in ascending order and the two fully associative designs are near the top as well. From this comparison one can see that the fully associative designs generally have a higher hit rate compared to the lower n way designs but when the n way designs are at least 8, the rates are almost similar. That is similar between the LRU 8 way and LRU fully associative and then the FIFO 8 way and FIFO fully associative.

As for cache size when looking at the bar graph one can see the gradual slope that forms on the top as the hit rate gets higher and higher for each increase in cache size. The results vary between designs but every design gradually increases their hit rate with the increase in cache size. Another interesting thing to note is that at the largest cache size on this graph, all the designs are very close in their hit rate percentages with the direct map still being the lowest. The first three cache sizes show the biggest discrepancies between hit rates for all of the designs.