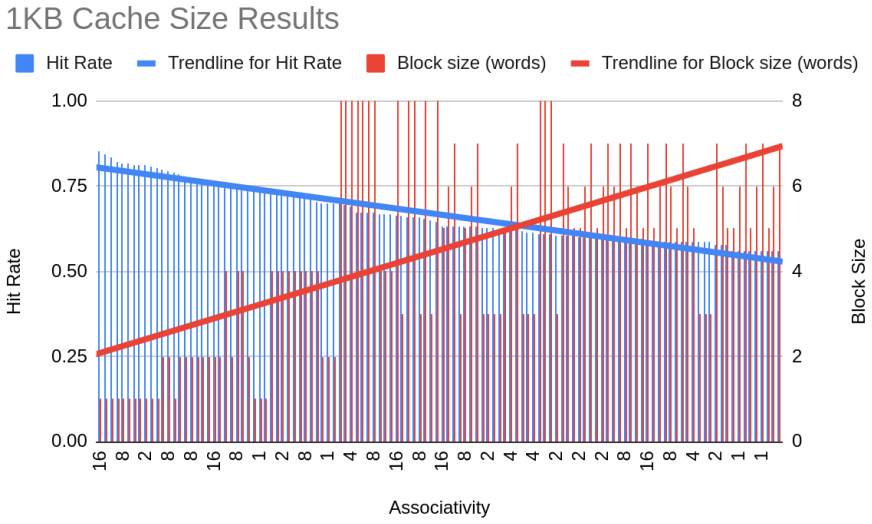
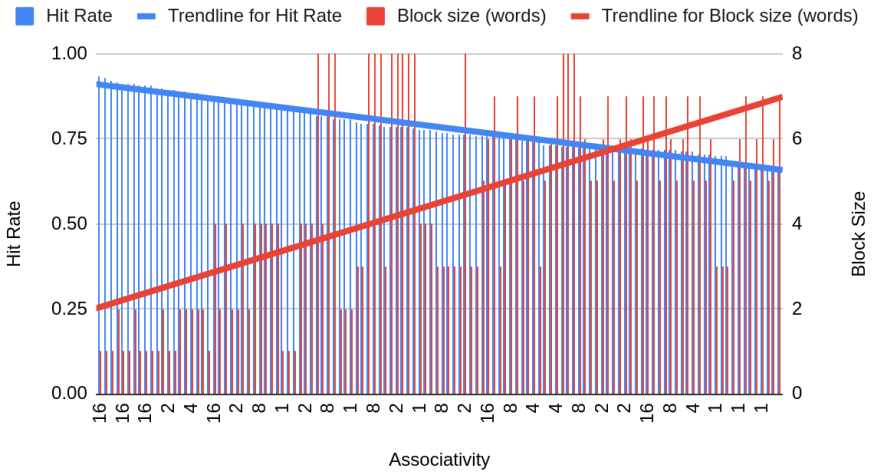
Nathan Panzer
Lab 6 (Cache Simulator) Writeup

In general, the best replacement policy is LRU, followed by NMRU+Random (pseudo-LRU), followed by Random. That's also the order of replacement policies in decreasing implementation difficulty. NMRU+Random seems to strike an effective balance between the two.

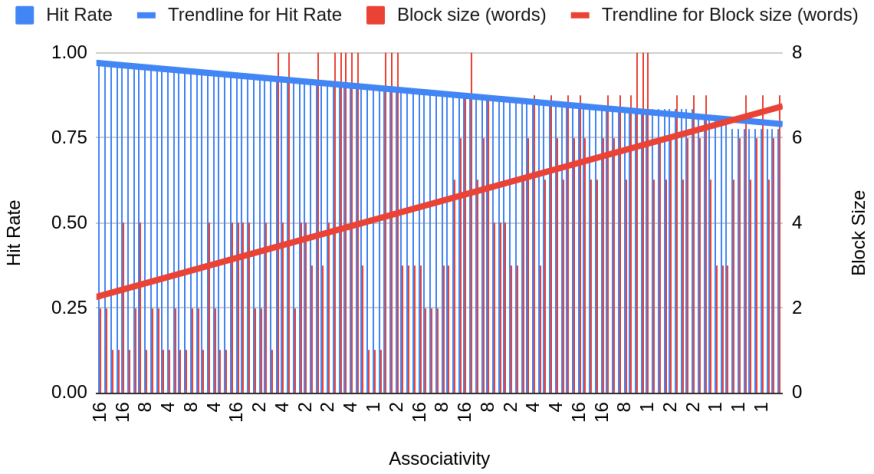
Cache Size	**Ideal Associativity**	**Ideal Block Size (words)**	**Hit Rate**	**Notes**
1KB	16-8	1	0.8516546667	Small block size leads to less cache pollution, which is a big deal in such a small cache.
2KB	16-4	1	0.9336386667	
4KB	16-4	2	0.9636553333	The cache is becoming big enough that we can increase the block size without getting too much cache pollution.
8KB	16-4	8-4	0.9830686667	In this dataset (the given trace file), there are a lot of similar addresses, so a larger block size is beneficial.
16KB	16-4	8-4	0.992922	
32KB	16-4	8-4	0.9955293333	
64KB	16-4	8-4	0.9969306667	
128KB	16-4	8	0.99815	
256KB	16-4	8	0.99815	At this point, such petty differences such as replacement policy don't matter, there's so much room in the cache.

These charts are a little weird because there are two independent variables (associativity and block size), but you can mostly see the hit rate trend as associativity and block size change.

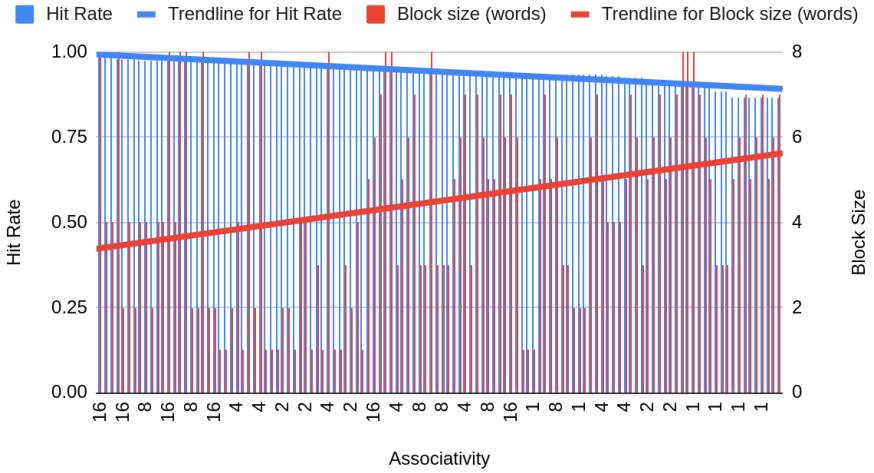




4KB Cache Size Results

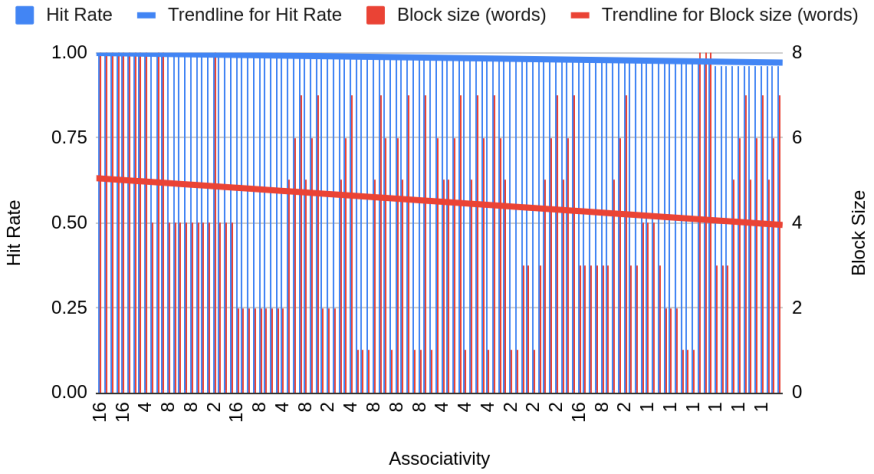


8KB Cache Size Results

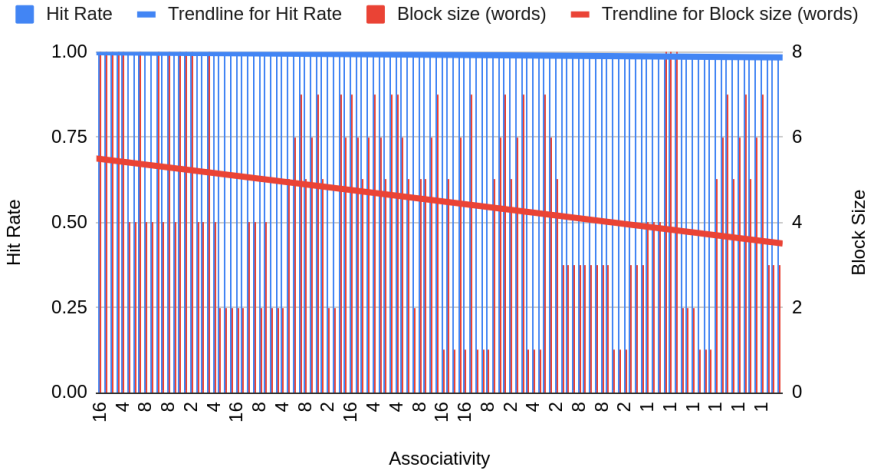


16KB Cache Size Results

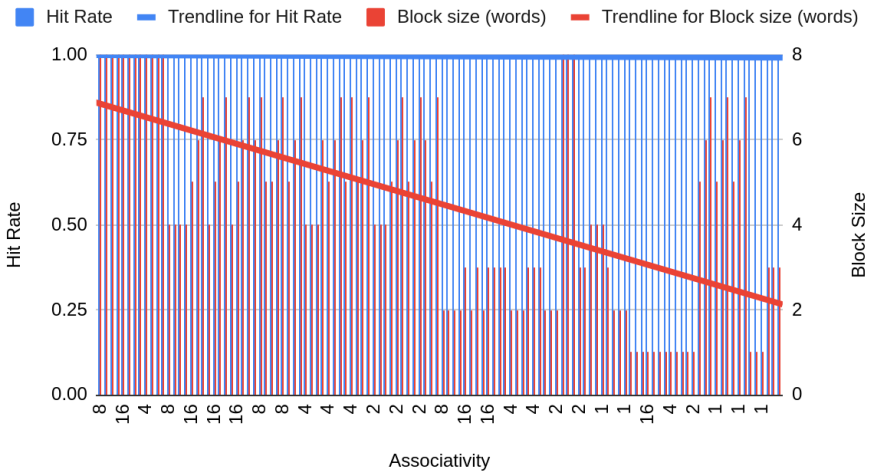
32KB Cache Size Results



64KB Cache Size Results



128KB Cache Size Results


256KB Cache Size Results
Hit Rate
Trendline for Hit Rate
Block size (words)
Trendline for Block size (words)
1.00
0.75
0.50
0.25
0.00
8
6
4
2
0
Hit Rate
Block Size
8 16 4 8 16 16 16 8 8 4 4 4 2 2 8 16 16 4 4 2 2 1 1 16 4 2 1 1 1
Associativity