Assignmet 4
Cache Size
Joshua Matthews

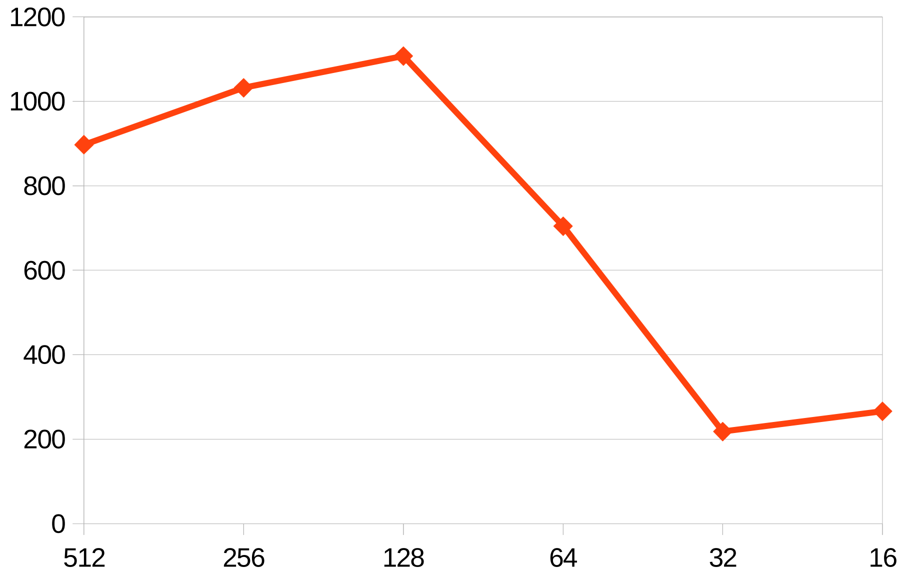
Cacheblock Design

The cacheblock program creates a large array of 4 million integers in memory. It then attempst to access various sections of the array several thousands of times in increasing jumps, measuring the time needed to access each part. The time after each attempted access is printed out, and used to determine the cache line size by looking for a sudden increase in access time that would indicate a miss. The jumps are measured in decreasing order instead of increasing, since for some reason whenever increasing values were attempted the program would produce a segmentation fault.

Cacheblock Results

Based on the results when run on assembly.rutgers.edu, the cache size on that machine is 64 bytes. This is indicated by a sudden drop in access times when the jump size drops below 64.

Graph of cacheblock resutls:



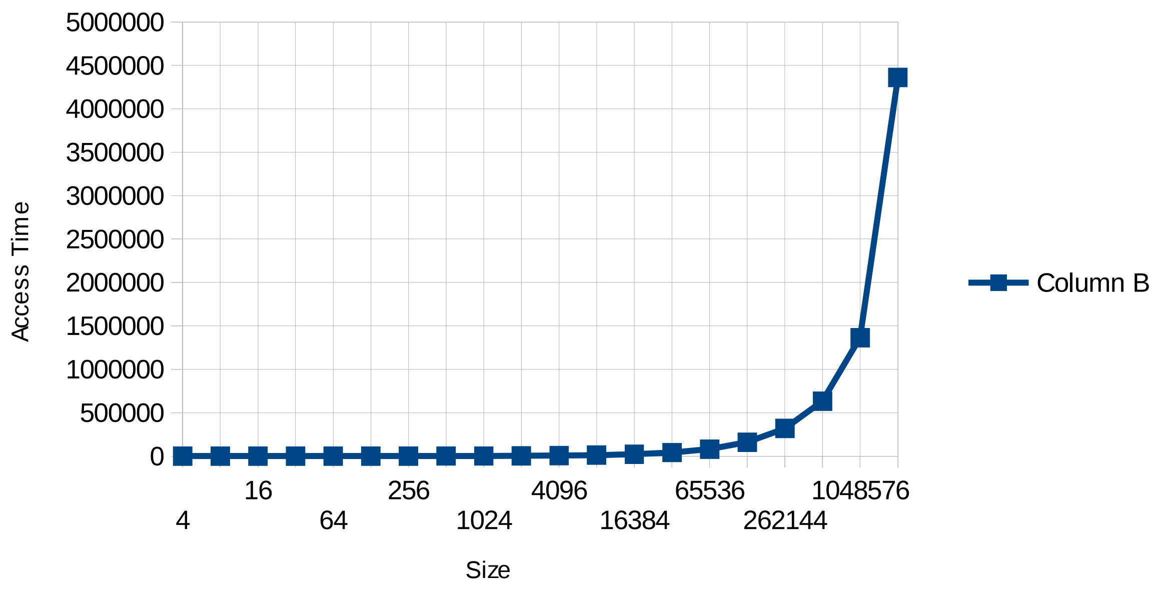
Cachesize Design

The cachesize program was designed similar to the cacheblock program. It loops over a large array, but jumps are made in 64 byte increments (since the line size was determined to be 64 bytes). The goal is to fill the cache and then eventually access data past the cache, producing a sudden jump in the access time.

Cachesize Results

The cachesize results were not as clear as clear as the cacheblock results. However, there appears to be a point where the times begin increasing faster at 6MB. This would indicate the the misses are starting to occur, so it is reasonable to conclude that the cache is 6MB in size.

Cachesize results graph:


Access Time
Size
Column B