Assignment 3

By:-

Vidhu Arora (IMT2021082)

Shlok Agrawal (IMT2021103)

Shreyan Gupta (IMT2021039)

Question 1 (Cache)

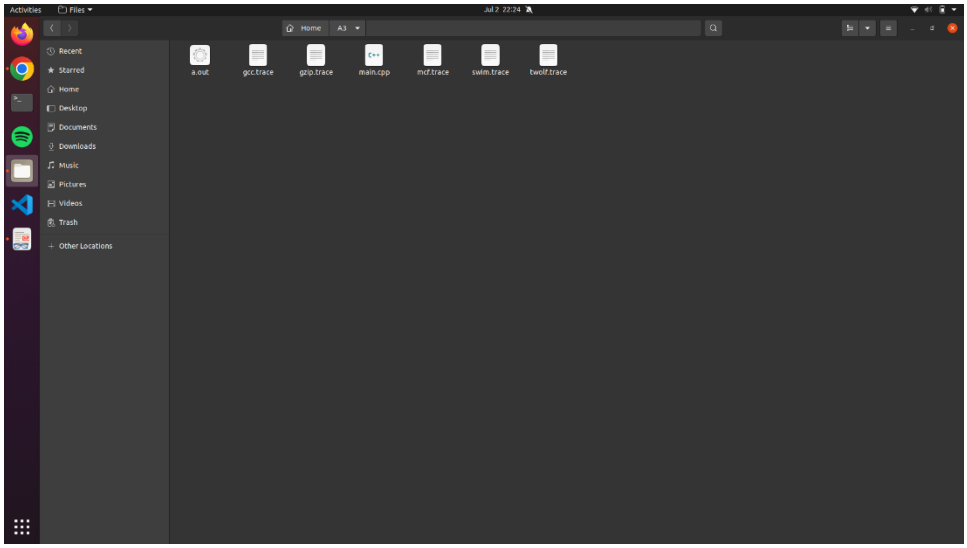
Q1(a)

- We have implemented the cache in the form of 2-D array. The block size, cache size and number of ways can be varied dynamically through input. Other values such as number of sets, number of blocks etc. are calculated by the program and the user need not worry about them.
- In the code, we created a structure called **block** which represents an individual block in cache. It has attributes valid, tag and time. Time attribute in a cache block contains the time at which the block was last accessed. For the values of time, we simply use the instruction number (first instruction has instruction number equal to 1, second instruction = 2 and so on).
- We have implemented LRU policy for replacement using the time field.

Instructions to run code

1.) Compile the main.cpp file using command g++ <roll_number>_main.cpp

2.) Place the extracted Trace files in the same folder as the a.out file generated in step 1. Your folder should look like the following:



3.) Run the a.out file using the command ./a.out

4.) You will be prompted to enter the cache size, block size and associativity of the cache. Enter the desired values and hit enter.

5.) Now the program will run and report the hit rate, hits and misses for the five traces files provided.

6.) Alternatively, if you are using an IDE, make sure that the trace file are in the same folder as the main.cpp file and run the code as you would normally run any code in your IDE. Sample Output:

```
vidhu@vidhu-Lenovo-Legion-5-15ARH05:~/OneDrive/Academic/Sem 2/Computer Architecture/assignment/assignment 3$ ./a.out
Enter cache size (in KB), block size (in Bytes) and associativity
512 4 4

input file: gcc.trace
Hit Rate = 93.8353720495%
Hits = 483894  ||  Misses = 31790

input file: gzip.trace
Hit Rate = 66.7056096623%
Hits = 320884  ||  Misses = 160161

input file: mcf.trace
Hit Rate = 1.03254674237%
Hits = 7509  ||  Misses = 719722

input file: swim.trace
Hit Rate = 92.622545301%
Hits = 280826  ||  Misses = 22368

input file: twolf.trace
Hit Rate = 98.7614560141%
Hits = 476845  ||  Misses = 5980

Program executed in 1.12333 seconds
vidhu@vidhu-Lenovo-Legion-5-15ARH05:~/OneDrive/Academic/Sem 2/Computer Architecture/assignment/assignment 3$
```

Q1(b)

Upon changing the cache size from 512KB to 2048KB, following observations are made:

1.) Hit rates increase slightly for gcc.trace.

2.) However, for the rest of the traces, the hit rates remain the same.

----------Varying Cache Size------- ----------

----------Comparing Hits---------- ----------

Cache Size	512KB	2048KB
gcc	483894	483895
gzip	320884	320884
mcf	7509	7509
swim	280826	280826
twolf	476845	476845

----------Comparing Hit Rate------- ----------

Cache Size	512KB	2048KB
gcc	93.8354	93.836
gzip	66.7056	66.7056
mcf	1.03255	1.03255
swim	92.6225	92.6225
twolf	98.7615	98.7615

Q1(c)

Upon increasing the block size, hit rates for all the trace files show an increasing trend.

mcf.trace shows the biggest jump in hit rate, going from 1.03846% to 50.5031% upon changing the block size from 8 to 16 Bytes.

----------Varying Block Size--------- ---------- ----------

----------Comparing Hits---------- ---------- ----------

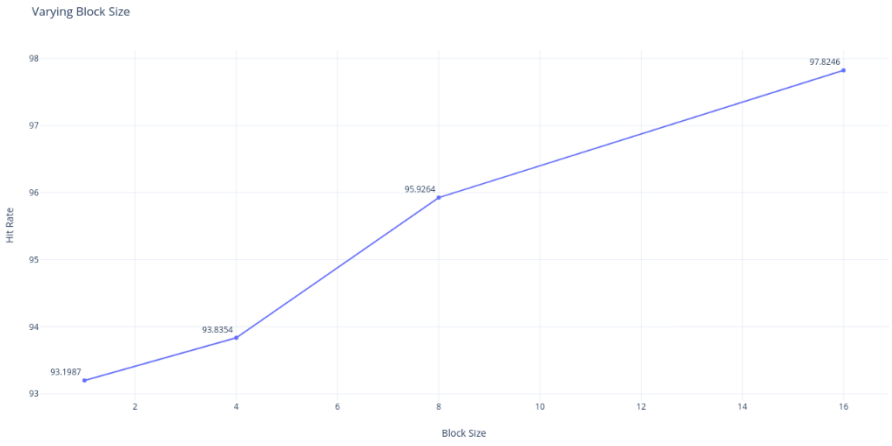
Block size	1B	4B	8B	16B
gcc	480611	483894	494677	504466
gzip	320876	320884	320892	321269
mcf	7452	7509	7552	367274
swim	280589	280826	283378	291771
twolf	475471	476845	477320	479870

----------Comparing Hit Rates------ ---------- ----------

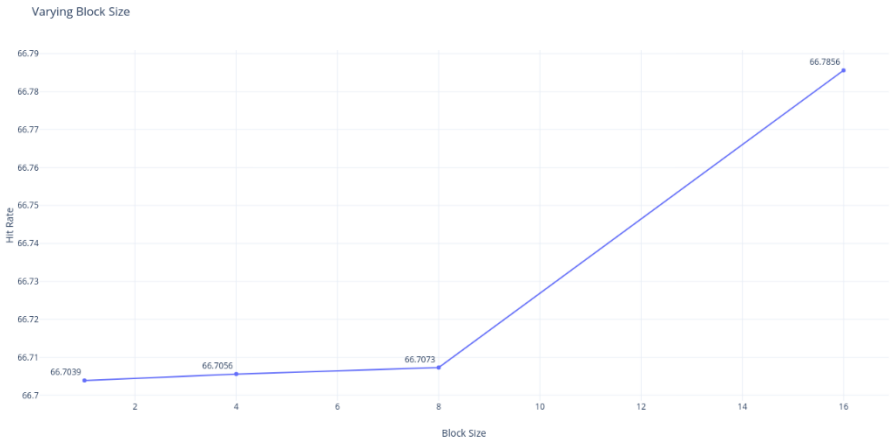
Block size	1B	4B	8B	16B
gcc	93.1987	93.8354	95.9264	97.8246
gzip	66.7039	66.7056	66.7073	66.7856
mcf	1.02471	1.03255	1.03846	50.5031
swim	92.5444	92.6225	93.4643	96.2324
twolf	98.4769	98.7615	98.8598	99.388

The following graphs illustrate this data:

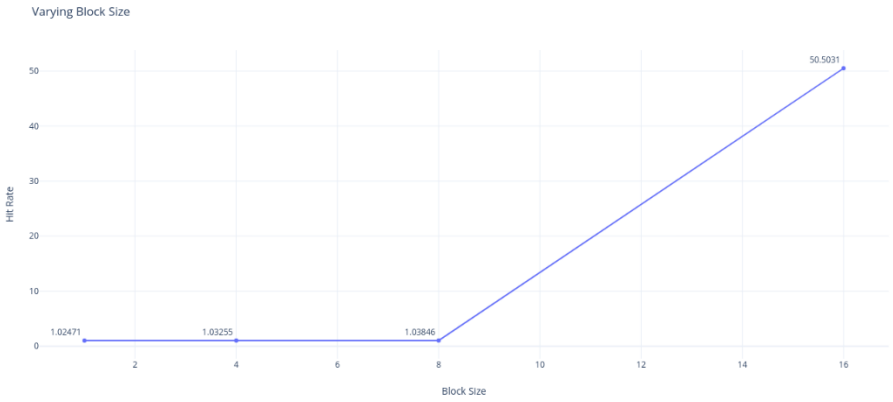
gcc.trace



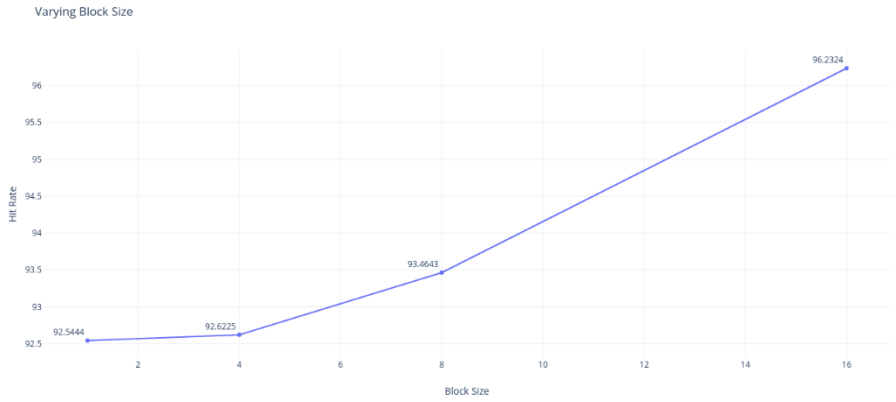
gzip.trace



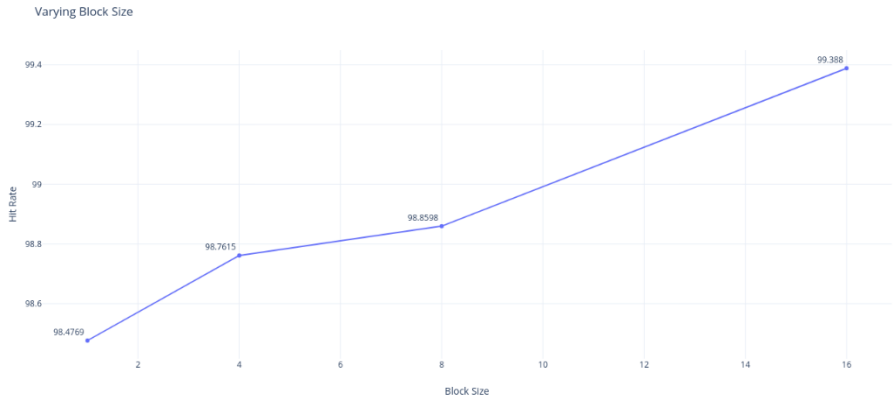
mcf.trace



swim.trace



twolf.trace



Q1(d)

Upon increasing the associativity, there is an increase in the hit rate up to a certain value, after which the hit rate seems to plateau. gzip.trace shows no change in hit rate upon increasing associativity.

----------Varying Associativity----------

----------Comparing Hits----------

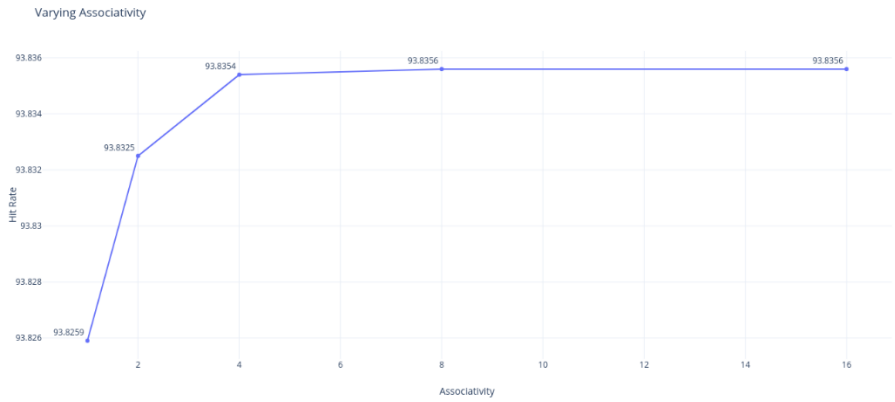
Sets	1	2	4	8	16
gcc	483845	483879	483894	483895	483895
gzip	320884	320884	320884	320884	320884
mcf	7506	7508	7509	7509	7509
swim	280739	280826	280826	280826	280826
twolf	476772	476842	476845	476845	476845

----------Comparing Hit Rates----------

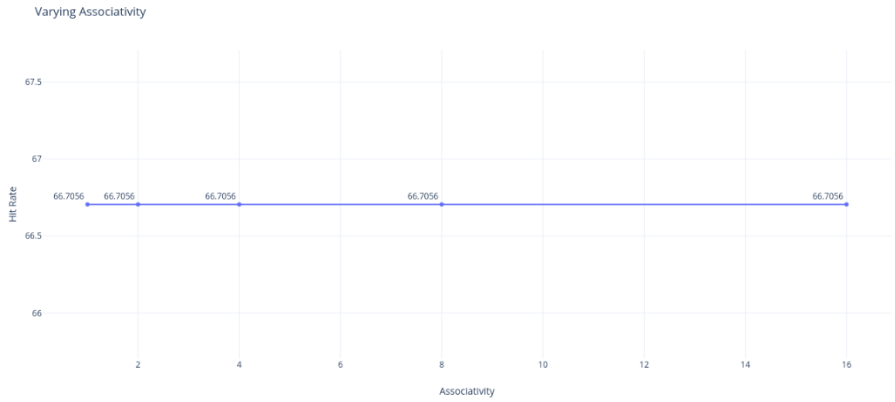
Sets	1	2	4	8	16
gcc	93.8259	93.8325	93.8354	93.8356	93.8356
gzip	66.7056	66.7056	66.7056	66.7056	66.7056
mcf	1.03213	1.03241	1.03255	1.03255	1.03255
swim	92.5939	92.6225	92.6225	92.6225	92.6225
twolf	98.7463	98.7608	98.7615	98.7615	98.7615

The following graphs illustrate this data:

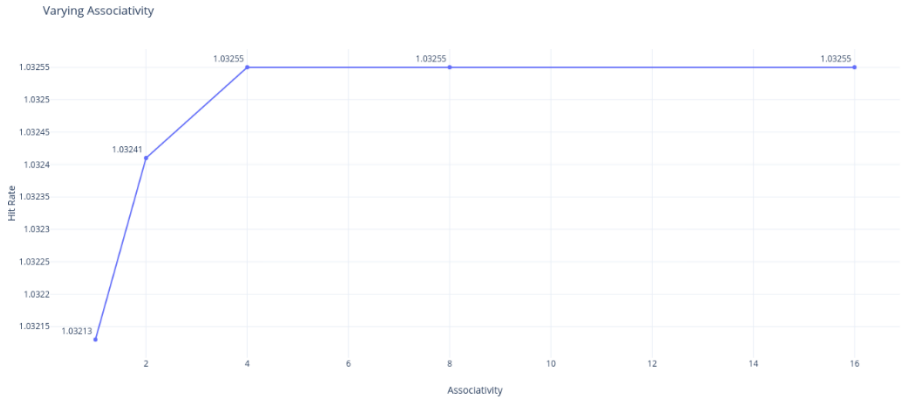
gcc.trace



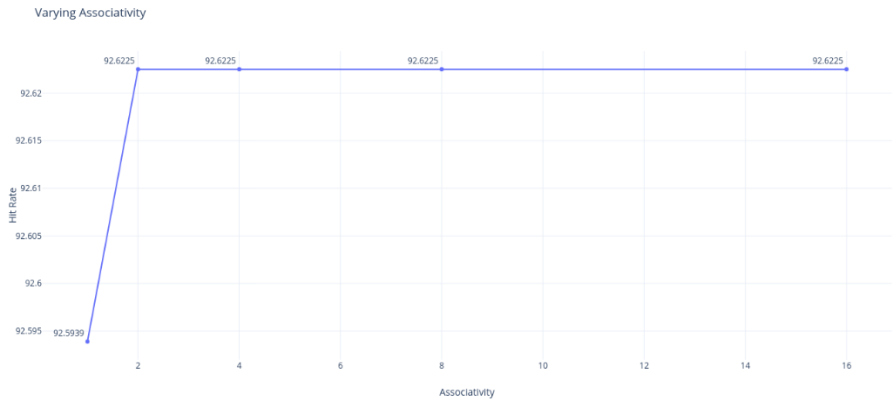
gzip.trace



mcf.trace



swim.trace



twolf.trace