Assignment 3

Output of the program ➔

```
1. gcc.trace

Accesses : 515683

Direct Cache
=> Hits    : 483504
=> Misses : 32179
=> Hit   Rate : 0.937599261561851
=> Miss Rate : 0.06240073843814902
=> Hit To Miss Rate : 15.025451381335653

Set Associative Cache
=> Hits    : 483871
=> Misses : 31812
=> Hit   Rate : 0.9383109390846702
=> Miss Rate : 0.06168906091532977
=> Hit To Miss Rate : 15.21032943543317

2. gzip.trace

Accesses : 481044

Direct Cache
=> Hits    : 320883
=> Misses : 160161
=> Hit   Rate : 0.6670554044952229
=> Miss Rate : 0.3329445955047771
=> Hit To Miss Rate : 2.003502725382584

Set Associative Cache
=> Hits    : 320883
=> Misses : 160161
=> Hit   Rate : 0.6670554044952229
=> Miss Rate : 0.3329445955047771
=> Hit To Miss Rate : 2.003502725382584

3. mcf.trace

Accesses : 727230

Direct Cache
=> Hits    : 7505
=> Misses : 719725
=> Hit   Rate : 0.01031998129890131
=> Miss Rate : 0.9896800187010987
=> Hit To Miss Rate : 0.010427593872659697

Set Associative Cache
=> Hits    : 7508
=> Misses : 719722
=> Hit   Rate : 0.010324106541259299
=> Miss Rate : 0.9896758934587407
=> Hit To Miss Rate : 0.01043180561383423
```

```
=> Hit To Miss Rate : 2.003502725382584

3. mcf.trace

Accesses : 727230

Direct Cache
=> Hits    : 7505
=> Misses : 719725
=> Hit   Rate : 0.01031998129890131
=> Miss Rate : 0.9896800187010987
=> Hit To Miss Rate : 0.010427593872659697

Set Associative Cache
=> Hits    : 7508
=> Misses : 719722
=> Hit   Rate : 0.010324106541259299
=> Miss Rate : 0.9896758934587407
=> Hit To Miss Rate : 0.01043180561383423

4. swim.trace

Accesses : 303193

Direct Cache
=> Hits    : 280738
=> Misses : 22455
=> Hit   Rate : 0.9259382637461947
=> Miss Rate : 0.07406173625380533
=> Hit To Miss Rate : 12.502248942329103

Set Associative Cache
=> Hits    : 280825
=> Misses : 22368
=> Hit   Rate : 0.9262252096849202
=> Miss Rate : 0.07377479031507983
=> Hit To Miss Rate : 12.554765736766809

5. twolf.trace

Accesses : 482824

Direct Cache
=> Hits    : 476770
=> Misses : 6054
=> Hit   Rate : 0.9874612695309264
=> Miss Rate : 0.012538730469073616
=> Hit To Miss Rate : 78.75289065080938

Set Associative Cache
=> Hits    : 476844
=> Misses : 5980
=> Hit   Rate : 0.9876145344887578
=> Miss Rate : 0.012385465511242191
=> Hit To Miss Rate : 79.73979933110368
architsangal@Archit-Sangal:~/CA last assignment/code/code
```

Important Points to keep in mind:

1) The structure of the code is very delicate. Code files are in folder named as "code". The trace files are in folder named as "trace". There is an extra file in "trace" folder with the name "test.ply". This can we used for custom test cases if you don't want to edit other ".trace" files. For using the test.trace file read the comments in the 'main()' of the class "Main".

2) Main logic is in 2 function "Set_Associative_Cache" and "Direct_Cache". Other function are helping them to work properly.

3) I have used the LRU (least recently used) replacement policy for the replacement in set associative cache. The element which will be replaced will have the "**lowest recent number**". Recent number is not a standard terminology. We are using it here to explain things.

Recent Number	Implication
4	Recently used
3	2nd most recently used
2	3rd most recently used
1	Least Recently Used
0	Element not assigned

4) **Commands To Be Used** :-
   >> javac DirectCache.java Main.java SetAssociativeCache.java
   >> java Main

```
architsangal@Archit-Sangal:~/CA last assignment/code/code$ javac DirectCache.java Main.java SetAssociativeCache.java
architsangal@Archit-Sangal:~/CA last assignment/code/code$ java Main
```

   Note: You should be in the directory which has these files –
   a) DirectCache.java
   b) Main.java
   c) SetAssociativeCache.java

5) Output in a form of table – (Rates and ratio are being rounded off or truncated)

Direct Cache –

	gcc.trace	gzip.trace	mcf.trace	swim.trace	twolf.trace
Number of Accesses	515683	481044	727230	303193	482824
Number of Hits	483504	320883	7505	280738	476770
Misses	32179	160161	719725	22455	6054
Hit Rate	0.9376	0.6671	0.0103	0.9259	0.9875
Miss Rate	0.0624	0.3329	0.9897	0.0741	0.0125
Hit to Miss Ratio	15.0255	2.0035	0.0104	12.5022	78.7529

4-way Set Associative Cache –

	gcc.trace	gzip.trace	mcf.trace	swim.trace	twolf.trace
Number of Accesses	515683	481044	727230	303193	482824
Number of Hits	483871	320883	7508	280825	476844
Misses	31812	160161	719722	22368	5980
Hit Rate	0.9383	0.6671	0.0103	0.9262	0.9876
Miss Rate	0.0617	0.3329	0.9897	0.0738	0.0124
Hit to Miss Ratio	15.2103	2.0035	0.0104	12.5548	79.7398

Observations

1) For all the input files –

Hit rates of 4-way set associative >= Hit rates of direct mapped cache.
Hit To Miss ratio of 4-way set associative >= Hit To Miss ratio of direct mapped cache.
Miss rates of 4-way set associative <= Miss rates of direct mapped cache.

These observations goes well hand in hand with "spatial locality" of memory.

2)   Logically 4-way set associative cache is very complex than that of direct mapped cache. That felt true while coding.

Output (in Text form)-

1. gcc.trace

Accesses : 515683

Direct Cache

=> Hits   : 483504

=> Misses : 32179

=> Hit Rate : 0.937599261561851

=> Miss Rate : 0.06240073843814902

=> Hit To Miss Rate : 15.025451381335653

Set Associative Cache

=> Hits   : 483871

=> Misses : 31812

=> Hit Rate : 0.9383109390846702

=> Miss Rate : 0.06168906091532977

=> Hit To Miss Rate : 15.21032943543317

2. gzip.trace

Accesses : 481044

Direct Cache

=> Hits   : 320883

=> Misses : 160161

=> Hit Rate : 0.6670554044952229

=> Miss Rate : 0.3329445955047771

=> Hit To Miss Rate : 2.003502725382584

Set Associative Cache

=> Hits   : 320883

=> Misses : 160161

=> Hit Rate : 0.6670554044952229

=> Miss Rate : 0.3329445955047771

=> Hit To Miss Rate : 2.003502725382584

3. mcf.trace

Accesses : 727230

Direct Cache

=> Hits   : 7505

=> Misses : 719725

=> Hit Rate : 0.01031998129890131

=> Miss Rate : 0.9896800187010987

=> Hit To Miss Rate : 0.010427593872659697

Set Associative Cache

=> Hits   : 7508

=> Misses : 719722

=> Hit Rate : 0.010324106541259299

=> Miss Rate : 0.9896758934587407

=> Hit To Miss Rate : 0.01043180561383423

4. swim.trace

Accesses : 303193

Direct Cache

=> Hits   : 280738

=> Misses : 22455

=> Hit Rate : 0.9259382637461947

=> Miss Rate : 0.07406173625380533

=> Hit To Miss Rate : 12.502248942329103

Set Associative Cache

=> Hits   : 280825

=> Misses : 22368

=> Hit Rate : 0.9262252096849202

=> Miss Rate : 0.07377479031507983

=> Hit To Miss Rate : 12.554765736766809

5. twolf.trace

Accesses : 482824

Direct Cache

=> Hits   : 476770

=> Misses : 6054

=> Hit Rate : 0.9874612695309264

=> Miss Rate : 0.012538730469073616

=> Hit To Miss Rate : 78.75289065080938

Set Associative Cache

=> Hits   : 476844

=> Misses : 5980

=> Hit Rate : 0.9876145344887578

=> Miss Rate : 0.012385465511242191

=> Hit To Miss Rate : 79.73979933110368