Report

By --> Ajay Chhajed (IMT2019006)

How to Run code

a) I am taking input of a file name as a command line argument and taking one input file at a time.

b) So, you have to first compile your cpp code (directMap or setAssociative) and then an executable is created "./a.out".

c) Now, write command "./a.out file_name ", where file_name is equal to the name of the file such as "gcc.trace".

d) For example to run code for input file "gzip.trace" :

—> g++ -std=c++11 IMT2019006_setAssociative.cpp

—> ./a.out gzip.trace

Direct Mapped cache

FILE	TOTAL	HITS	MISSES	HITRATE	MISSRATE
gcc.trace	515683	483504	32179	0.937599	0.062400
gzip.trace	481044	320883	160161	0.667055	0.332945
mcf.trace	727230	7505	719725	0.010319	0.989680
swim.trace	303193	280738	22455	0.925938	0.074062
twolf.trace	482824	476770	6054	0.987461	0.012539

Set Associative cache

FILE	TOTAL	HITS	MISSES	HITRATE	MISSRATE
gcc.trace	515683	483871	31812	0.938311	0.0616891
gzip.trace	481044	320883	160161	0.667055	0.332945

mcf.trace	727230	7508	719722	0.0103241	0.989676
swim.trace	303193	280825	22368	0.926225	0.0737748
twolf.trace	482824	476844	5980	0.987615	0.0123855

Oberservation

There is not much difference in hit and miss rate according to my replacement policy used. I used replacement policy LRU least recent used, in this the tag which is least recently used among 4 tags is evicted (updated). According to this policy, Set Associative Cache will have a little greater hit rate and have a little smaller miss rate as compared to Direct Map Cache.

Results for Direct Map Cache(input/output):

```
ajaychhajed@Ajays-MacBook-Air ca % g++ -std=c++11 IMT2019006_directMap.cpp
ajaychhajed@Ajays-MacBook-Air ca % ./a.out gcc.trace
Number of Instructions : 515683
Number of hits : 483504
Number of miss : 32179
Hit rate : 0.937599
Miss rate : 0.0624007
ajaychhajed@Ajays-MacBook-Air ca % ./a.out gzip.trace
Number of Instructions : 481044
Number of hits : 320883
Number of miss : 160161
Hit rate : 0.667055
Miss rate : 0.332945
ajaychhajed@Ajays-MacBook-Air ca % ./a.out mcf.trace
Number of Instructions : 727230
Number of hits : 7505
Number of miss : 719725
Hit rate : 0.01032
Miss rate : 0.98968
ajaychhajed@Ajays-MacBook-Air ca % ./a.out swim.trace
Number of Instructions : 303193
Number of hits : 280738
Number of miss : 22455
Hit rate : 0.925938
Miss rate : 0.0740617
ajaychhajed@Ajays-MacBook-Air ca % ./a.out twolf.trace
Number of Instructions : 482824
Number of hits : 476770
Number of miss : 6054
Hit rate : 0.987461
Miss rate : 0.0125387
ajaychhajed@Ajays-MacBook-Air ca %
```

Results for Set Associative Cache :

```
ajaychhajed@Ajays-MacBook-Air ca % g++ -std=c++11 IMT2019006_setAssociative.cpp
ajaychhajed@Ajays-MacBook-Air ca % ./a.out gcc.trace
Number of Instructions : 515683
Number of hits : 483871
Number of miss : 31812
Hit rate : 0.938311
Miss rate : 0.0616891
ajaychhajed@Ajays-MacBook-Air ca % ./a.out gzip.trace
Number of Instructions : 481044
Number of hits : 320883
Number of miss : 160161
Hit rate : 0.667055
Miss rate : 0.332945
ajaychhajed@Ajays-MacBook-Air ca % ./a.out mcf.trace
Number of Instructions : 727230
Number of hits : 7508
Number of miss : 719722
Hit rate : 0.0103241
Miss rate : 0.989676
ajaychhajed@Ajays-MacBook-Air ca % ./a.out swim.trace
Number of Instructions : 303193
Number of hits : 280825
Number of miss : 22368
Hit rate : 0.926225
Miss rate : 0.0737748
ajaychhajed@Ajays-MacBook-Air ca % ./a.out twolf.trace
Number of Instructions : 482824
Number of hits : 476844
Number of miss : 5980
Hit rate : 0.987615
Miss rate : 0.0123855
ajaychhajed@Ajays-MacBook-Air ca %
```