1. Tables

trace file: simple

memsize: 50

	HitRate	HitCount	MissCount	TotalEvict	CleanEvict	DirtyEvict
rand	71.6555	7670	3034	2984	272	2712
fifo	72.1880	7727	2977	2927	204	2723
lru	73.9817	7919	2785	2735	89	2646
clock	73.8322	7903	2801	2751	97	2654
opt	74.9626	8024	2680	2630	29	2601

memsize: 100

	HitRate	HitCount	MissCount	TotalEvict	CleanEvict	DirtyEvict
rand	74.1125	7933	2771	2671	64	2607
fifo	74.2059	7943	2761	2661	44	2617
lru	74.8786	8015	2689	2589	2	2587
clock	74.8505	8012	2692	2592	3	2589
opt	75.3457	8065	2639	2539	0	2539

memsize: 150

	HitRate	HitCount	MissCount	TotalEvict	CleanEvict	DirtyEvict
rand	74.5422	7979	2725	2575	21	2554
fifo	74.5889	7984	2720	2570	16	2554
lru	74.9066	8018	2686	2536	0	2536
clock	74.8972	8017	2687	2537	0	2537
opt	75.3457	8065	2639	2489	0	2489

memsize: 200

	HitRate	HitCount	MissCount	TotalEvict	CleanEvict	DirtyEvict
rand	74.6357	7989	2715	2515	15	2500
fifo	74.6637	7992	2712	2512	12	2500
lru	74.9066	8018	2686	2486	0	2486
clock	74.8972	8017	2687	2487	0	2487
opt	75.3363	8064	2640	2440	0	2440

trace file: matmul

memsize: 50

	HitRate	HitCount	MissCount	TotalEvict	CleanEvict	DirtyEvict
rand	66.4848	1973461	994827	994777	955498	39279
fifo	62.0226	1841009	1127279	1127229	1083234	43995
lru	64.9218	1927065	1041223	1041173	1040067	1106
clock	64.9215	1927058	1041230	1041180	1040068	1112
opt	80.2086	2380823	587465	587415	586329	1086

memsize: 100

	HitRate	HitCount	MissCount	TotalEvict	CleanEvict	DirtyEvict
rand	89.0829	2644237	324051	323951	316281	7670
fifo	63.4956	1884731	1083557	1083457	1061230	22227
lru	66.0930	1961832	1006456	1006356	1005274	1082
clock	66.2500	1966492	1001796	1001696	1000612	1084
opt	96.8737	2875489	92799	92699	91615	1084

memsize: 150

	HitRate	HitCount	MissCount	TotalEvict	CleanEvict	DirtyEvict
rand	96.7449	2871668	96620	96470	94096	2374
fifo	98.8406	2933874	34414	34264	32947	1317
lru	98.8920	2935400	32888	32738	31657	1081
clock	98.8317	2933610	34678	34528	33446	1082
opt	99.1033	2941672	26616	26466	25383	1083

memsize: 200

	HitRate	HitCount	MissCount	TotalEvict	CleanEvict	DirtyEvict
rand	98.0846	2911434	56854	56654	55006	1648
fifo	98.8582	2934396	33892	33692	32436	1256
lru	98.8924	2935412	32876	32676	31595	1081
clock	98.8920	2935399	32889	32689	31608	1081
opt	99.3509	2949022	19266	19066	17983	1083

trace file: blocked

```
trace file:  blocked

memsize:  50

          HitRate    HitCount   MissCount  TotalEvict  CleanEvict  DirtyEvict
  rand     99.6714     2516774        8298        8248        5691        2557
  fifo     99.7448     2518629        6443        6393        4118        2275
   lru     99.7967     2519939        5133        5083        2748        2335
 clock     99.7726     2519331        5741        5691        3251        2440
   opt     99.8534     2521370        3702        3652        2568        1084

memsize:  100

          HitRate    HitCount   MissCount  TotalEvict  CleanEvict  DirtyEvict
  rand     99.7930     2519844        5228        5128        3404        1724
  fifo     99.8291     2520757        4315        4215        2734        1481
   lru     99.8501     2521286        3786        3686        2603        1083
 clock     99.8343     2520888        4184        4084        2608        1476
   opt     99.8812     2522072        3000        2900        1829        1071

memsize:  150

          HitRate    HitCount   MissCount  TotalEvict  CleanEvict  DirtyEvict
  rand     99.8242     2520633        4439        4289        2788        1501
  fifo     99.8331     2520858        4214        4064        2640        1424
   lru     99.8507     2521302        3770        3620        2558        1062
 clock     99.8502     2521290        3782        3632        2570        1062
   opt     99.9000     2522547        2525        2375        1300        1075

memsize:  200

          HitRate    HitCount   MissCount  TotalEvict  CleanEvict  DirtyEvict
  rand     99.8484     2521244        3828        3628        2289        1339
  fifo     99.8746     2521905        3167        2967        1868        1099
   lru     99.8536     2521375        3697        3497        2435        1062
 clock     99.8733     2521872        3200        3000        1938        1062
   opt     99.9099     2522797        2275        2075        1009        1066
```

2. Compression:
   hit rate compression table:

```
simpleloop
              50         100         150         200
  rand      71.6555     74.1125     74.5422     74.6357
  fifo      72.1880     74.2059     74.5889     74.6637
  lru       73.9817     74.8786     74.9066     74.9066
  clock     73.8322     74.8505     74.8972     74.8972
  opt       74.9626     75.3457     75.3457     75.3363

matmul
              50         100         150         200
  rand      66.4848     89.0829     96.7449     98.0846
  fifo      62.0226     63.4956     98.8406     98.8582
  lru       64.9218     66.0930     98.8920     98.8924
  clock     64.9215     66.2500     98.8317     98.8920
  opt       80.2086     96.8737     99.1033     99.3509

blocked
              50         100         150         200
  rand      99.6714     99.7930     99.8242     99.8484
  fifo      99.7448     99.8291     99.8331     99.8746
  lru       99.7967     99.8501     99.8507     99.8536
  clock     99.7726     99.8343     99.8502     99.8733
  opt       99.8534     99.8812     99.9000     99.9099
```

from the table above we could observe that ‘opt’ has the best performance, which proved the fact that ‘opt’ is an optimal solution. For each specific as swapfile size increases solution the hit rate it as well increases. In ‘matmul’, ’rand’ has a better performance than ‘fifo’, ‘lru’ and ‘clock’ but other than ‘matul’ ‘fifo’, ‘lru’ and ‘clock’ all has a better performance than ‘rand’. The performance of ‘lru’ and ‘clock’ are really close and especially for in ‘matul’ their performance is almost the same. In ‘simpleloop’, ‘lru’ and ‘matmul’ is a little better and in ‘block’, ’clock’ is a little better. Generally, the hit rate is like ‘fifo’ $<$ ‘lru’ = ‘clock’ $<$ ‘opt’.

3. LRU:
   LRU hit rate table:

	50	100	150	200
simpleloop	73.9817	74.8786	74.9066	74.9066
matmul	64.9218	66.0930	98.8920	98.8924
blocked	99.7967	99.8501	99.8507	99.8536
mine	99.8351	99.9305	99.9501	99.9501

   The Least-Recently- Used (LRU) policy replaces the least-recently-used page. From the hit rate table we could see that as swapfile size increases, the hit rate also increases. And LRU has a stable performance on 'simpleloop' around 75%. LRU has the worst performance on 'matmul' for $s = 50, 100$. LRU has the best performance on 'block' which has a stable hit rate around 99%. The performance of LRU goes above 95% when size > 150. Generally, LRU has a stable performance on 'loop' and 'block', and the performance on 'block' is better. The performance on 'matmul' varies a lot as size increases.