tr-matmul.ref

memsize=50	rand	lru	fifo	clock	opt
Hit rate	65.5518	63.9466	60.9672	63.9456	79.6588
Hit count	1893151	1846792	1760748	1846764	2300565
Miss count	994873	1041232	1127276	1041260	587459
Total references	2888024	2888024	2888024	2888024	2888024
Clean evictions	955573	1040070	1083233	1040095	586320
Dirty evictions	39250	1112	43993	1115	1089

memsize=100	rand	lru	fifo	clock	opt
Hit rate	88.7936	65.1506	62.4813	63.9456	96.7868
Hit count	2564380	1881566	1804476	1846764	2795225
Miss count	323644	1006458	1083548	1041260	92799
Total references	2888024	2888024	2888024	2888024	2888024
Clean evictions	315957	1005277	1061225	1040095	91611
Dirty evictions	7587	1081	22223	1115	1088

memsize=150	rand	lru	fifo	clock	opt
Hit rate	96.6629	98.8612	98.8085	98.85	99.0784
Hit count	2791649	2855136	2853613	2854811	2861408
Miss count	96375	32888	34411	33213	26616
Total references	2888024	2888024	2888024	2888024	2888024
Clean evictions	93851	31657	32944	31979	25378
Dirty evictions	2374	1081	1317	1084	1088

memsize=200	rand	lru	fifo	clock	opt
Hit rate	98.0379	98.8612	98.8085	98.85	99.0784
Hit count	2831358	2855136	2853613	2854811	2861408
Miss count	56666	32888	34411	33213	26616
Total references	2888024	2888024	2888024	2888024	2888024
Clean evictions	54832	31657	32944	31979	25378
Dirty evictions	1634	1081	1317	1084	1088

tr-simpleloop_heap.ref

memsize=50	rand	lru	fifo	clock	opt
Hit rate	71.0078	72.9845	71.1047	72.8391	74.1182
Hit count	7328	7532	7338	7517	7649
Miss count	2992	2788	2982	2803	2671
Total references	10320	10320	10320	10320	10320
Clean evictions	231	91	214	102	20
Dirty evictions	2711	2647	2718	2651	2601

memsize=100	rand	lru	fifo	clock	opt
Hit rate	73.0523	73.9729	73.2655	73.9341	74.3798
Hit count	7539	7634	7561	7630	7676
Miss count	2781	2686	2759	2690	2644
Total references	10320	10320	10320	10320	10320
Clean evictions	62	2	45	5	0
Dirty evictions	2619	2584	2614	2585	2494

memsize=150	rand	lru	fifo	clock	opt
Hit rate	73.7016	73.9922	73.6628	73.9632	74.3798
Hit count	7606	7636	7602	7633	7676
Miss count	2714	2684	2718	2687	2644
Total references	10320	10320	10320	10320	10320
Clean evictions	14	0	16	0	0
Dirty evictions	2550	2534	2552	2537	2494

memsize=200	rand	lru	fifo	clock	opt
Hit rate	73.6628	73.9922	73.7403	73.9826	74.3798
Hit count	7602	7636	7610	7635	7676
Miss count	2718	2684	2710	2685	2644
Total references	10320	10320	10320	10320	10320
Clean evictions	11	0	12	0	0
Dirty evictions	2507	2484	2498	2485	2444

tr-blocked.ref

memsize=50	rand	lru	fifo	clock	opt
Hit rate	99.6575	99.784	99.7314	99.7821	99.8465
Hit count	2409879	2412937	2411664	2412890	2414448
Miss count	8281	5223	6496	5270	3712
Total references	2418160	2418160	2418160	2418160	2418160
Clean evictions	5697	2817	4182	2872	2574
Dirty evictions	2534	2356	2264	2348	1088

memsize=100	rand	lru	fifo	clock	opt
Hit rate	99.7848	99.8434	99.8205	99.8338	99.8754
Hit count	2412957	2414372	2413819	2414140	2415148
Miss count	5203	3788	4341	4020	3012
Total references	2418160	2418160	2418160	2418160	2418160
Clean evictions	3394	2606	2760	2618	1836
Dirty evictions	1709	1082	1481	1302	1076

memsize=150	rand	lru	fifo	clock	opt
Hit rate	99.8188	99.8441	99.8252	99.8369	99.8954
Hit count	2413779	2414389	2413932	2414215	2415630
Miss count	4381	3771	4228	3945	2530
Total references	2418160	2418160	2418160	2418160	2418160
Clean evictions	2762	2559	2653	2575	1300
Dirty evictions	1469	1062	1425	1220	1080

memsize=200	rand	lru	fifo	clock	opt
Hit rate	99.8406	99.8471	99.8686	99.868	99.9057
Hit count	2414305	2414462	2414982	2414968	2415880
Miss count	3855	3698	3178	3192	2280
Total references	2418160	2418160	2418160	2418160	2418160
Clean evictions	2313	2436	1878	1928	1012
Dirty evictions	1342	1062	1100	1064	1068

tr-simpleloop_stack.ref

memsize=50	rand	lru	fifo	clock	opt
Hit rate	94.3824	95.7941	94.5	95.5441	97.2059
Hit count	6418	6514	6426	6497	6610
Miss count	382	286	374	303	190
Total references	6800	6800	6800	6800	6800
Clean evictions	152	88	149	98	18
Dirty evictions	180	148	175	155	122

memsize=100	rand	lru	fifo	clock	opt
Hit rate	97.0147	97.4265	97.1176	97.3971	97.5588
Hit count	6597	6625	6604	6623	6634
Miss count	203	175	196	177	166
Total references	6800	6800	6800	6800	6800
Clean evictions	11	0	0	1	0
Dirty evictions	92	75	96	76	66

memsize=150	rand	lru	fifo	clock	opt
Hit rate	97.5294	97.5441	97.4265	97.4559	97.5588
Hit count	6632	6633	6625	6627	6634
Miss count	168	167	175	173	166
Total references	6800	6800	6800	6800	6800
Clean evictions	0	0	0	0	0
Dirty evictions	18	17	25	23	16

memsize=200	rand	lru	fifo	clock	opt
Hit rate	97.5588	97.5588	97.5588	97.5588	97.5588
Hit count	6634	6634	6634	6634	6634
Miss count	166	166	166	166	166
Total references	6800	6800	6800	6800	6800
Clean evictions	0	0	0	0	0
Dirty evictions	0	0	0	0	0

Comparison :

The fourth program we choose the run is the simpleloop that uses stack memory instead of heap memory. The program has the second highest hit rate amoung all the programs when the memory size is small, and the memory size does not really influence the hit rate of the program (The hit rate is usually around 97% for all memory size). Interestingly, the hit rate of stack simple loop is identical for all algorithm when the memory size is 200. When the memory size increases, the hit rate of matmul will overrun the hit rate of the simple stack loop. The heap simple loop program has the lowest hit rate when the memory size increase to 150, 200, and the memory size does not influence the hit rate of heap simple loop like the other programs (such as matmul). Also, the stack simple loop performs way better than the heap simple loop in all algorithm amoung all memory size.

When the memory size (size of the coremap) is the same, comparing all the algorithm, OPT usually has the highest hit rate since it “predicts” (read the tracefile ahead) to choose which one to evict. Then the hit rate follows as LRU, Clock, FIFO and Rand in decreasing order. The total eviction number is Rand, FIFO, Clock, LRU, OPT in increasing order when their memsize is the same. The hit rate of Rand algorithm might sometimes higher than Clock, FIFO and LRU when running matmul and with small memory size (50 and 100). One chararacteristic that all the algorithm shares are that, as the memory size increases, the hit rate and hit count increases respectively , and all the evition count (Total reference in our table) decreases due to the decrease in needs of swapping.

Description of LRU algorithm :

LRU algorothm will evict the least frequently used frame, which we indicated as the frame that was referenced in the longest time ago compare to other frame. The LRU algorithm performs very badly when running matmul with small memory size such as 50 and 100. However, LRU algorithm performs well with matmul when the memory size increases to 150 and 200. When the memory size changes from 100 to 150, the hit rate of LRU with matmul program increases more than 20%. In comparison with other algorithm, we notice that for simple algorithm like LRU and FIFO, its hit rate increases as the memory size of matmul program increases from 100 to 150. One of the possible reason of this weird behavior is that the total number of memory addresss in matmul program is between 100 to somewhere around 150, so that the coremap(physical memory) does not need to really evict many pages often, the memory frames are sufficient to store almost all the memory address in matmul program. Therefore, despite the few miss count when the program first referece a page, we rarely need to evict pages when the memory size is big enough to store all the memory address.