Algorithem FIFO Tracefile: tr-simpleloop.ref(2048 trace)

Memory Size	Hit count	Miss count	Clean Evic	Dirty Evic
50	7248	2960	199	2711
100	7454	2754	44	2610
150	7493	2715	16	2549
200	7510	2707	12	2495

Algorithem Clock

Memory Size	Hit count	Miss count	Clean Evic	Dirty Evic
50	7428	2780	90	2640
100	7523	2685	3	2582
150	7526	2684	0	2532
200	7526	2682	0	2482

Algorithem FIFO Tracefile: tr-blocked.ref(2048 trace)

Memory Size	Hit count	Miss count	Clean Evic	Dirty Evic
50	2411635	6493	4174	2269
100	24139984	4224	2759	1476
150	2413904	4224	2653	1421
200	2414956	3172	1876	1096

Tracefile: tr-blocked.ref(2048 trace)

Algorithem Clock

Memory Size	Hit count	Miss count	Clean Evic	Dirty Evic
50	2412371	5757	3278	2429
100	2413821	4307	2613	1594
150	2414349	3779	3779	1058
200	2414920	3208	1941	1067

Tracefile: tr-mutulref(2048 trace)

Algorithem FIFO Memory Size	Hit count	Miss count	Clean Evic	Dirty Evic
50	1760586	1127262	1083226	43986
100	1804305	1083543	1061221	22222
150	2853440	34408	32944	1314
200	2853961	33887	32434	1253

Algorithem Clock Tracefile: tr-mutual.ref(2048 trace)

Memory Size	Hit count	Miss count	Clean Evic	Dirty Evic
50	1846624	1041224	1040069	1105
100	1886055	1001793	1000614	1079
150	2853134	34714	33485	1079
200	2854961	32887	31609	1078

Total referentces	Hit rate	Miss rate
10208	71.0031	28.997
10208	73.0212	26.979
10208	73.4832	26.5968
10208	73.4816	26.5184

Total referentces	Hit rate	Miss rate
10208	72.7667	27.2335
10208	73.6971	26.3029
10208	73.7265	26.2736
10208	73.7265	26.2735

Total referentces	Hit rate	Miss rate
2418128	99.7315	0.2685
2418128	99.82	0.1793
2418128	99.8253	0.1747
2418128	99.8688	0.1312

Total referentces	Hit rate	Miss rate
2418128	99.7619	0.2381
2418128	99.8219	0.1781
2418128	99.8437	0.1563
2418128	99.8673	0.1327

Total referentces	Hit rate	Miss rate
2887848	60.9653	39.0347
2887848	62.4792	37.5208
2887848	98.8085	1.1915
2887848	98.8266	1.1734

Total referentces	Hit rate	Miss rate
2887848	63.9446	36.0554
2887848	65.3101	34.6899
2887848	98.7979	1.2021
2887848	98.8612	1.1388

Program I choice

The fourth program I choice is system call tree, I found out that it didn't go through the memory process since hit and miss are both 0.

Best algorithm

The best algorithms must be OPT since OPT is able to precast the future memory usage, then it must has the highest hit rate, more over CLOCK and LRU should really competitive to each other since there are many unsure about the memory usage in the future. The FIFO is the slowest algorithm among them all, but since the memory size increase it still has reasonable hit rate.

Paragraph 2:

Since the memory is increasing hit rate for both FIFO and LRU are getting higher especially for FIFO algorithm. However, since the memory keep increasing the increment for hit rate become really slow. The marginal increase rate for bot algorithm is small when they has sufficient space.