CS40 HW7 UM Labnotes
Ivi Fung (sfung02) and Matthew Wong (mwong14)

Stage Progression Table
Room 240

Benchmark	Fetches	Time (sec)	Rel to Start	Rel to Prev
Starting Point UM HW6 Submission				
midmark	62735811727	14.08	1.000	1.000
advent	[redacted]	121.73	1.000	1.000
sandmark	[redacted]	351.59	1.000	1.000
Stage 1 Compiled with optimization turned on to O1 and linked against -lcii-O1				
midmark	50021168361	11.22	0.797	0.797
advent	[redacted]	95.54	0.785	0.785
sandmark	[redacted]	272.54	0.775	0.775
Stage 2 Compiled with optimization turned on to O2 and linked against -lcii-O2				
midmark	44391667470	9.67	0.687	0.862
advent	[redacted]	83.11	0.682	0.870
sandmark	[redacted]	241.33	0.686	0.885
Stage 3 Replaced Bitpack with actual bit math (flags etc.)				
midmark	33156289537	8.27	0.587	0.855
advent	[redacted]	74.03	0.608	0.890
sandmark	[redacted]	206.63	0.588	0.856
Stage 4 Cached the size of the program segment (segment 0) and only changed it to be a uint32_t that only updates on initialization or in loadprogram (as the size only updates then)				
midmark	17010185338	3.88	0.276	0.469
advent	[redacted]	32.27	0.265	0.436

sandmark	[redacted]	97.33	0.277	0.471
Stage 5 Removed assertions for failure modes				
midmark	9803909669	2.13	0.151	0.549
advent	[redacted]	16.04	0.132	0.497
sandmark		56.27	0.160	0.578
Stage 6 Removed static inline (to see qcachegrind breakdown)				
midmark	9422873161	2.04	0.145	0.958
advent	[redacted]	15.18	0.125	0.946
sandmark		51.68	0.147	0.918
Stage 7 Made the CPU_State struct a global variable (Undid change)				
midmark	9478798293	2.08	0.147	1.020
advent	[redacted]	15.81	0.130	1.041
sandmark		52.53	0.149	1.016
Stage 8 Removed Hanson Sequences from Memory.c and replaced it with an array that doubles in size at max capacity				
midmark	6012192381	0.93	0.066	0.447
advent	[redacted]	7.45	0.061	0.471
sandmark		24.15	0.069	0.460
Stage 9 Moved register calls inside of the function calls				
midmark	538680501	0.96	0.068	1.032
advent	[redacted]	7.48	0.061	1.004
sandmark		24.50	0.070	1.014
Stage 10 Moved Memory into CPU.c				
midmark	5086077504	0.88	0.063	0.917

advent	[redacted]	6.93	0.057	0.926
sandmark		23.00	0.065	0.939
Stage 11 Moved Segment into CPU.C				
midmark	4712528284	0.71	0.050	0.795
advent	[redacted]	6.00	0.049	0.866
sandmark		18.68	0.05	0.812
Stage 12 Removed get_instructions, placed relevant code into the main switch statement				
midmark	4542387244	0.68	0.048	0.958
advent	[redacted]	5.48	0.045	0.913
sandmark		17.86	0.051	0.956
Stage 13 Removed segment functions, so the functions directly reference the segment				
midmark	4450637640	0.64	0.045	0.941
advent	[redacted]	5.28	0.043	0.963
sandmark		16.04	0.045	0.898
Stage 14 Static inline'd execute_function (our main switch statement)				
midmark	3040914940	0.56	0.040	0.875
advent	[redacted]	4.06	0.033	0.770
sandmark		14.00	0.040	0.873
Stage 15 Moved Memory module outside of CPU_State struct				
midmark	2984643732	0.51	0.036	0.911
advent	[redacted]	3.82	0.031	0.941
sandmark		12.80	0.036	0.914
Stage 16 Made CPU state into variables in the run function (moving execute function into the run function)				

midmark	2351951562	0.40	0.028	0.784
advent	[redacted]	3.26	0.027	0.853
sandmark	[redacted]	10.21	0.029	0.798
Stage 17 Made Mem free, map, and unmap functions static inline				
midmark	2258470727	0.38	0.027	0.950
advent	[redacted]	2.89	0.024	0.887
sandmark	[redacted]	9.79	0.028	0.959
Stage 18 Moved everything into main				
midmark	2238674388	0.37	0.026	0.974
advent	[redacted]	2.83	0.023	0.979
sandmark	[redacted]	9.56	0.027	0.977
Stage 19 Added *mainProgram variable to specifically store pointer to main memory				
midmark	2194773499	0.36	0.026	0.973
advent	[redacted]	2.69	0.022	0.951
sandmark	[redacted]	9.26	0.026	0.969