Labnotes - Homework 7

Phong Hoang and Christian Zinck

Tufts University

Here is our record of UM emulator's performance after five stages of changing and improving our source code in homework 6.

Table 1: Report on performance of UM emulator

Benchmark	Time	Instructions	Rel to start	Rel to prev	Improvement
midmark.um	2.61s	2.0219×10^{10}	1.000	1.000	No improvement (starting point)
advent.umz	22.63s		1.000	1.000	
sandmark.umz	64.98s		1.000	1.000	
midmark.um	2.24s	1.8615×10^{10}	0.858	0.858	Compiling with -O2
advent.umz	20.17s		0.903	0.903	
sandmark.umz	55.76s		0.858	0.858	
midmark.um	2.06s	1.512×10^{10}	0.789	0.920	Reconstructed main loop variables and masks
advent.umz	18.45s		0.826	0.915	
sandmark.umz	51.63s		0.795	0.926	
midmark.um	1.66s	1.282×10^{10}	0.636	0.806	Removed run_op function, put all arithmetic from operations in main function, converted switch statement into priority if-else tower
advent.umz	13.75s		0.616	0.745	
sandmark.umz	41.35s		0.636	0.801	
midmark.um	0.50s	4.968×10^{9}	0.192	0.301	Implemented sequence ADT to replece all Hanson sequences and adjusted um struct pointer and segment to use new sequence and store unint32_t* use new instead of void*
advent.umz	4.25s		0.190	0.309	
sandmark.umz	12.62s		0.194	0.305	
midmark.um	0.32s	3.515×10^{9}	0.123	0.640	Moved segment load and segment store into main instead of calling functions
advent.umz	2.61s		0.117	0.614	
sandmark.umz	7.94s		0.122	0.630	