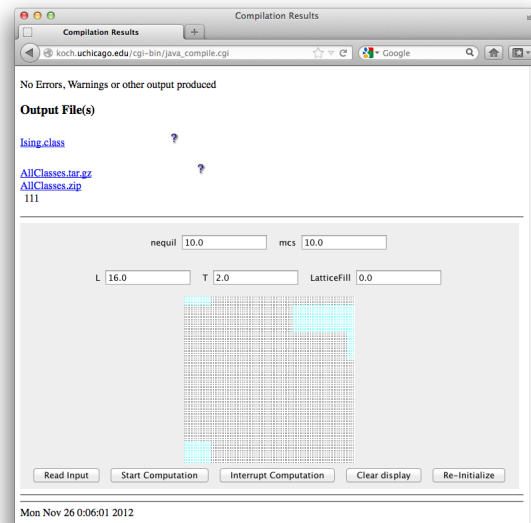


Java Console						
T	accept	Eave	E2ave	Mave	M2ave	absMave
02	0.236	-1.288	426.150	-0.101	3.723	0.101
02	0.196	-1.336	457.856	-0.014	2.013	0.077
02	0.230	-1.278	422.375	-0.022	2.022	0.073
02	0.200	-1.416	514.425	0.010	1.695	0.066
02	0.203	-1.383	491.106	-0.397	41.156	0.397
02	0.083	-1.717	756.881	-0.905	210.420	0.905
02	0.071	-1.786	818.681	-0.935	224.258	0.935
02	0.077	-1.770	803.319	-0.923	218.581	0.923
02	0.066	-1.783	815.581	-0.930	221.923	0.930
02	0.070	-1.791	822.450	-0.927	220.495	0.927
02	0.068	-1.770	805.144	-0.930	221.553	0.930
02	0.089	-1.648	697.006	-0.890	202.930	0.890
02	0.071	-1.752	787.669	-0.913	213.619	0.913
02	0.060	-1.831	859.763	-0.948	230.444	0.948
02	0.129	-1.583	642.456	-0.784	157.981	0.784
02	0.198	-1.388	494.400	-0.326	27.948	0.326
02	0.214	-1.330	453.094	-0.241	15.170	0.241
02	0.132	-1.614	668.456	-0.644	106.588	0.644
02	0.087	-1.730	767.081	-0.891	204.111	0.891
02	0.111	-1.636	690.756	-0.832	179.070	0.832
02	0.155	-1.494	574.075	-0.763	150.620	0.763
02	0.056	-1.830	858.231	-0.948	230.027	0.948
02	0.061	-1.778	810.675	-0.918	216.005	0.918
02	0.109	-1.647	696.300	-0.878	197.666	0.878
02	0.075	-1.763	796.088	-0.916	215.230	0.916
02	0.112	-1.648	697.194	-0.875	196.116	0.875
02	0.072	-1.767	801.781	-0.920	217.138	0.920
02	0.093	-1.713	752.588	-0.905	210.167	0.905
02	0.052	-1.852	878.856	-0.958	234.981	0.958
02	0.061	-1.831	859.575	-0.948	230.419	0.948
02	0.081	-1.806	836.200	-0.937	224.792	0.937
02	0.073	-1.773	806.581	-0.931	222.169	0.931
02	0.077	-1.711	751.544	-0.909	211.495	0.909
02	0.085	-1.709	749.888	-0.907	210.739	0.907
02	0.219	-1.364	479.481	0.498	65.728	0.498
02	0.105	-1.730	771.019	0.874	197.523	0.874
02	0.075	-1.734	771.563	0.921	217.352	0.921
02	0.045	-1.842	869.156	0.955	233.739	0.955
02	0.050	-1.833	861.156	0.953	232.691	0.953
02	0.091	-1.738	773.688	0.901	207.870	0.901
02	0.091	-1.723	760.856	0.917	215.413	0.917
02	0.057	-1.789	820.444	0.940	226.236	0.940
02	0.126	-1.598	656.794	0.821	173.442	0.821
02	0.077	-1.706	746.788	0.894	204.788	0.894
02	0.092	-1.709	751.275	0.888	202.636	0.888
02	0.081	-1.767	800.556	0.920	216.959	0.920
02	0.075	-1.772	804.738	0.927	220.289	0.927
02	0.047	-1.827	854.631	0.951	231.464	0.951



So far, I have managed to port the BASIC code from Gould&Tobochnik into java source code which runs and visualizes the Ising model using the Metropolis algorithm. The code is completely contained in `Ising.java`.

At the moment, system parameters such as temperature, initial spin config, and system size can be set by the user along with simulation parameters regarding the number of monte carlo steps per data point.

My code can calculate and output system properties such as energy and magnetization. Additionally, it can display the system on a 256x256 pixel grid (as long as  $L = 2^n \leq 256$ ). There seems to be one slight trouble with this visualization (at the high end of either coordinate, there is only half a cell). However, this should be easily remedied with a review of my source code.

The next step is to implement the user-chosen measurement capabilities for central boxes in the system, as well as more thorough verification of the accuracy of my code.