CPSC 335 – Team AJA – Project #1 Cellular Automaton Rule 45

Alexandria Wolfram

Juan Sanchez

Andrew Pham

For Project #1 the task was to display Wolfram's Cellular Automaton Rule 45 acting on a grid of 400 by 400 dimensions using Javascript. Our solution to this was to display the values using an array that would color squares on a grid in the correct pattern. To create a display that modeled Wolfram's Cellular Automaton Rule 45 we had to iterate through the values in the array initialize each value as either 0 or 1 in compliance with Cellular Automaton Rule 45. After the values are initialized, they are then displayed on a grid as squares colored either white for a zero value or black for a one value. Therefore, because we had to iterate through each generation and then each of its cells, the Big O running time of the program is $O(n \times m)$ where n would the number of generations and m would be the number of cells in each row. In this program the number of generations is equal to the number of cells in each row so the Big O running time would be $O(n^2)$.

To model the rules we first created an array to hold the values. The first 400 elements of this array were filled with all zeros except for in the 200^{th} position where a one value was placed. The next values are then populated with values dictated by Cellular Automaton Rule 45 using if statements and a nested for loop. To display the values graphically a single for loop was used to iterate through and color the squares either white for a zero value or black for a one value. Because the array is iterated through first using a double for loop and then using a single for loop for display the running time would appear to be $O(n^2+n)$ however the n is eventually overshadowed by the n^2 so our true Big O running time would be $O(n^2)$.