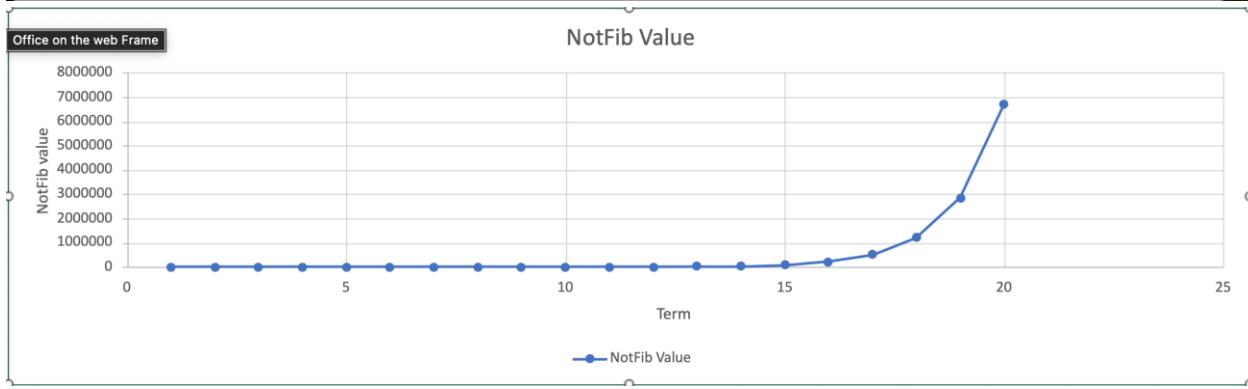


The screenshot shows a LeetCode problem page for "Remove Element". The problem statement asks to remove all occurrences of a value `val` from an array `nums` in-place. The code editor shows a Java implementation that has passed 116/116 test cases. The runtime is 0 ms, which is 100.00% faster than other submissions. The sidebar on the right shows the user's profile, "Justin_Ho09", and various navigation links like My Lists, Notebook, Progress, Points, Try New Features, Orders, My Playgrounds, Settings, Appearance, and Sign Out.



This plot only shows the first 20 terms in the sequence, and with that the exponential growth of this sequence can be seen. But there are some problems. The first one is that the difference between the first 15 terms is pretty much indistinguishable, and it looks like the not fib value there is 0 when it is not in reality. For example, term 1 is 0 and term 2 is 2, but you cannot tell at this scale. This issue is caused by the distance between each subsequent term growing larger each time. Since they're growing larger and creating greater distance between the current term and previous one, then eventually it will get to a point where only the current term is visibly displaced off the x axis. Another issue is that the growth of the sequence will reach the integer limit before it even reaches the first 1200; it does not even generate the result for the 100th term. So those are the problems encountered when trying to plot the first 1200 numbers of the not Fibonacci sequence.