Mental Analysis and How it was Critical in Developing my Program

When I was first at the drawing board for my program, I first wrote down in my notebook that I needed to solve for the probability that, based on a certain number of people, two people shared a birthday. I was also told that I needed to make an approximation of the probability over a given number of trials, which meant to me that I needed another way of calculating the probability and that this additional method had to be included into the program. Mental analysis is critical here because in noting that there are certain variables that help determine the expected outcome (being in the amount of people and the number of trials/experiments), it helps me determine the parameters of my methods and what I should expect to be passing through as arguments in my program. In knowing what parameters I am setting for my methods and in understanding what my overall outcome should be, I can now venture into each method and into figuring out how I can assign random individuals a birthday. While it may seem easier to randomly assign everyone a specific birth month and birthdate, it takes mental analysis to realize that specifying everyone a birth month and birthdate is a lot of code, and that if we assign everyone a random number between 1 – 365, we are able to provide everyone a randomized birthday without creating extra variables. This also demonstrates how mental analysis is critical in developing my program because it can help you save time and become more efficient. Overall, this lab has shown me that while I can translate every aspect of a problem into a program to create an answer, what separates a better programmer from a worse one is how the better programmer is able to make their thoughts more concise and still provide a program that represents all of the aspects of the problem, without having unnecessary added code.