HW01

Prasun Surana
CS5008, Summer 1

In the worse case, how many guesses would it our guessing game take to get the right answer if we had no hints at all? Explain.

Let us assume that the random number generated was 1. The maximum number of guesses to arrive at (i.e. the worst case) would be if we started at 10, then reduced our guess incrementally by 1 to 9, then 8, then 7, then 6, and so on, until we arrived at 1. Therefore, the worst case scenario would be 10 guesses to arrive at the correct number. This is analogous to the linear search algorithm, since we sequentially went through each element in the list.

In the worst case, how many guesses does it take to get the right number if we get a hint of "higher or lower" when guessing numbers 1-10 and guess intelligently (always picking in the middle of the remaining set of numbers)?

Let us assume that once again, the random number generated is 1, and let us also assume that with every iteration, we are told that the number is lower than our guess. Since we have 10 numbers to choose from and we intelligently pick the midpoint for each guess, our first guess would be 5. We are told that the actual number is lower, so we can discard the set of values from 6-10. Again choosing the midpoint, our next guess would be 2.5, but since we are dealing only with integers, we round it up to 3. We are again told that the number is lower than 3, so we can discard 4 and 5. Take the midpoint again, we get 1.5, but we round it up to 2, and discard 3. We again take the midpoint, 1, which is the correct value. We can see that the maximum number of guesses to arrive at the correct value is 4.

This process does not decrease linearly, but logarithmically, since we halve the set of possible values with each iteration. We know from the definition of logarithms that $\log_2 x$ means 'how many times do we have to multiply 2 by itself to arrive at x?'. We have 10 possible values that we can guess, and using this binary search method, we need to find $\log_2 10$, or in other words, 'how many times do we have to multiply 2 by itself to arrive at 10?'. $\text{Log}_2 10$ is equal to approximately 3.32, but since we cannot have a fraction of a guess, we round it up to 4, mathematically confirming what we observed if we manually applied binary search to the set of values from 1-10.