Noah Krill
Data Structures
Feb 4 2020
Algorithm 1 theoretical runtime is O(N). This holds true with the empirical analysis because when the N was set to 1,000,000 the time was 0.001147 and when it was bumped up to 10,000,000 it took 0.014375. 1,000,000/0.001447= 10,000,000/10(0.001447). The first algorithm had one for loop to iterate through so its O(N).

Algorithm 2 theoretical runtime is O(N^2). This holds true with the empirical analysis because when the N was set to 10,000 the time was 0.145396 and when it was bumped up to 100,000 it took 14.5725. 10,000/0.145396 = 100,000/10(14.5725)^2. This algorithm had two sets of for loops causing this algorithm to be O(N^2).

Algorithm 3 theoretical runtime is O(N^3). This holds true with the empirical analysis because when the N was set to 200 the time was 0.014925 and when it was bumped up to 1,000 it took 2.65667. 200/0.014925 = 1,000 /5(0.014925)^3. This algorithm had two sets of for loops causing this algorithm to be O(N^3).

Algorithm 4 theoretical runtime is O(N^2). Algorithm 2 theoretical runtime is O(N^2). This holds true with the empirical analysis because when the N was set to 10,000 the time was 0.072802and when it was bumped up to 100,000 it took 7.24494. 10,000/ 0.072802= 100,000/10(0.072802)^2. This algorithm had two sets of for loops causing this algorithm to be O(N^2).

Algorithm 5 theoretical runtime is O(N^5). This holds true with the empirical analysis because when the N was set to 100 the time was 2.00145 and when it was bumped up to 200 it took 66.1488. 100/ 2.00145 = 200/2(2.00145)^5. This algorithm had three sets of for loops with a multiplication inside causing this algorithm to be O(N^5).

Algorithm 6 theoretical runtime is O(N^4). This holds true with the empirical analysis because when N=1 the time was 0.025785. When it was updated to 500 the time went up to 17.9082. 100/0.025785=500/5(0.025785)^4. This algorithm had three sets of for loops with a multiplication inside and an if statement causing this algorithm to be O(N^4).

Exercise 2.6

A. on the nth day it would cost ((2^2)^N-1)

D=2^2^N-1

Log D = 2^N-2

Log log D + 1 = N

B. O(loglog N) days

2.11

A. linear = .5 *5 = 2.5 ms

Noah Krill
Data Structures
Feb 4 2020
B. O(N logN) = 0.5 × 5 log2 5 = 5.8 ms

C. quadratic =0.5 × 52 = 12.5 ms

D. cubic = 0.5 × 53 = 62.5 ms