Erikson Sodergren

CSCI 261 HW 1

Problem 1.

1. logba=log87=0.93 n=Ω(n0.93+ε) case 3: T(n)=Θ(n)

7(n/8)<=cn ε>=.07, c= 7/8

1. logba=log42=1/2 =Θ(n1/2) case 2: T(n)=Θ(n1/2logn) ε=0
2. logba=log39=2 n2=Θ(n2) case 2: T(n)=Θ(n2logn) ε=0
3. logba=log24=2 nlog2n=O(n2-ε) case 1: T(n)=Θ(n2) ε<1

Problem 2.

1. The array used by my program consists of the data at index i being the subsequence of length i ending in the smallest possible value.
2. During runtime the data of array(i+1) is decided by copying the data from array(i) and then appending the new number that extends the sequence. array(i) can then be overwritten with a better sequence later however, so you can no longer determine array(i+1) using it once the program has completed.
3. The program prints the resulting longest increasing subsequence but does not return anything.

Problem 3.

1. The array used by my program consists of the number of ways to read the input from index i to the end, using only vowels.
2. If the input at index i is a dot following a dash or the third dash in an O, array(i)=array(i+x), where x is how many morse characters used in this letter. If the input at i is a dash otherwise, array(I)=0. If the input at i is a dot and is the first character read, array(i)=1. If the input at i is a dot following the completion of a letter involving a dash, array(i)=array(i+1). If the input at i is a dot following a dot which had not ended a letter involving a dash, array(i)=array(i+1)+array(i+2).
3. The program prints the final count of how many ways the morse input can be read, but does not return anything.