CS 101, Assignment 4

Christopher Huynh, Oliver Rene

March 2016

1 Question 1

There are two for loops that run inevitably n times. Therefore it runs in $n^2 + c$ times which is in $O(n^2)$

2 Question 3

hadmatmult(H,k):
  n=size(x)
  if(n==2)
    return x[0]-x[1],x[1]-x[0]
  A=split(H,n/2) $O(n)$
  block1=hadmatmult(A[0][0],x[0][n/2]) $T(\frac{n}{4})$
  block2=hadmatmult(A[0][0],x[n/2+1][n]) $T(\frac{n}{4})$
  return [block1+block2,block1-block2]

Time complexity $T(n) = 2 * T(\frac{n}{4}) + c * n$
This means $T(n) \leq T(\frac{n}{2})$
The total cost is $2^i(\frac{n}{4^i})$or $n(\frac{1}{2})^i$
Analyzing it down the tree yields $\sum_{i=0}^{MaxDepth} n(\frac{1}{2})^i$
$n * \sum_{i=0}^{MaxDepth}(\frac{1}{2})^i \leq n * \sum_{i=0}^{\infty}(\frac{1}{2})^i$
There the time complexity is $O(n)$. The previous, brute force method requires $O(n^2)$. Therefore it is much faster.

3 Question 5

The divide and conquer method proves to be faster in practice. This is in agreeance with our time complexity derivations.