Yu Long Wang Assignment 3

Q1: $(1+2+\cdots+n) \leq (n+n+\cdots+n)$

$\leq n^2$, $c=1$ $n_0=1$ $g(n)=n^2$

$f(n) \leq c \cdot g(n)$ for $n \geq n_0$

So this algo is $O(n^2)$

Q2: $A: \begin{array}{c|c} & n \\ k & \begin{bmatrix} 1 & \cdots & n \\ \vdots & \ddots & \\ k & & \end{bmatrix} \end{array}$ $B: \begin{array}{c|c} & k \\ m & \begin{bmatrix} 1 & \cdots & k \\ \vdots & & \\ m & & \end{bmatrix} \end{array}$

the run time will be $n \cdot m \cdot k$

assume n is the largest number such that $n \geq m \geq k$

so $n \cdot m \cdot k \leq n \cdot n \cdot n = n^3$

this algo is $O(n^3)$

Q3: a) $T(n) = T(n-1) + T(n-1) = 2T(n-1)$ $T(n-1) = 2T(n-2)$

$T(n) = 2T(n-2) + 2T(n-2)$

$T(n) = 4T(n-2)$

$T(n) = 2(Tn-1)$ $n-1$ 2^1

$T(n) = 4T(n-2)$ $n-2$ 2^2

$T(n) = 8T(n-3)$ $n-3$ 2^3

$\vdots$

$T(n) = 2^n T(0)$ 0

$T(n) = 2^n \cdot 1$

$\Rightarrow$ this algo is $O(2^n)$

$T(0) = 2^{-1} + 2^{-1}$

$2^0 = \frac{1}{2} + \frac{1}{2}$

$2^0 = 1 \Rightarrow T(0) = 1$

Q3 b) $0 \to n-1$, This algo is $O(n)$

Q4 $T(n) = 2T(n/2) + n$, $n > 1$ $T(1) = 1$

$T(\frac{n}{2}) = 2T(\frac{n}{4}) + \frac{n}{2}$

$\Rightarrow T(n) = 2 \cdot (2T(\frac{n}{4}) + \frac{n}{2}) + n$

$= 4T(\frac{n}{4}) + n + n$

$= 4T(\frac{n}{4}) + 2n$

$\Rightarrow T(n) = 4 \cdot (2T(\frac{n}{8}) + \frac{n}{4}) + 2n$

$= 8 \cdot T(\frac{n}{8}) + 3n$

$\Rightarrow$ $T(n) = 2T(\frac{n}{2}) + n$ 1

$T(n) = 4T(\frac{n}{4}) + 2n$ ⋮

$T(n) = 8T(\frac{n}{8}) + 3n$ ⋮

⋮

$T(n) = 2^k T(\frac{n}{2^k}) + kn$ k

let $\frac{n}{2^k} = 1$

$n = 2^k$

$k = \log_2 n$

$\Rightarrow T(n) = n \cdot T(1) + \log n \cdot n$, $T(1) = 1$

$= n + \log n \cdot n \leq \log n \cdot n + \log n \cdot n = 2n \log n \in O(n \cdot \log n)$