HOMEWORK 1 – Q4

MINGLANG XIE

z5228006

4. You are in an orchard which has a quadratic shape of size $4n$ by $4n$ with equally spaced trees. You purchased apples from n^2 trees which also form a square, but the owner is allowing to choose such a square anywhere in the orchard. You have a map with the number of apples on each tree. Your task is to choose such a square which contains the largest total number of apples and which runs in time $O(n^2)$. Note that the brute force algorithm would run in time $\theta(n^4)$. (20 points)

Solution:

First, for $M[0][0:n]$ (total number of apples of first row of first n^2 trees), we can computer it (name as $T[0][1]$) in $O(n^2)$ by simply iterating the matrix. Then for each $i \geq n$, we have $T[row][i] = T[row][i-1] - M[row][i-n+1] + M[row][i]$. Follow this algorithm to compute the whole matrix into a two dimension array ($3n+1\ by\ 4n$), so we can compute each subsequent $S[i]$ in $O(1)$ time, giving an $O(n^2)$ algorithm.

Second, for $T[0:n][0]$ (first n^2 trees), we can compute it in

$O(n^2)$ by simply iterating. Then, for each $j \geq n$, we have $S[j] = S[j-1] - T[row][j-n+1] + T[row][j]$, so we compute each subsequent $S[j]$ in $O(1)$ time, and find out the largest total number of apples, giving an $O(n^2)$ algorithm.

row n terms

col n term

M[1][1]	M[1][2]	…	…	…	…	…	M[1][4n]
M[2][1]	M[2][2]	…	…	…	…	…	M[2][4n]
…	…	…	…	…	…	…	…
…	…	…	…	…	…	…	…
…	…	…	…	…	…	…	…
…	…	…	…	…	…	…	…
…	…	…	…	…	…	…	…
M[4n][1]	…	…	…	…	…	…	M[4n][4n]

Calculation each n terms by first add up each row n terms, then add each col n term, it would form a n^2 trees that we want to calculate.