CS6033 Design and Analysis of Algorithms I
Homework Assignment 2
09.24.2022

Group Members:
rk4305 (Sai Rajeev Koppuravuri)
sg7372 (Sriharsha Gaddipati)
vt2184 (Veeravenkata Raghavendra Naveenkumar Tata)
vt2182 (Venu Vardhan Reddy Tekula)

---

1. (10 points) You have the following 3 statements

   (a) (3 points) Algorithm A, divides the original problem into 7 subproblems of size $n/3$, recursively solves the subproblems, and then solves the original problem by combining the subproblems in linear time. Provide recurrence formula and running time in Θ notation.

   Ans. A divides the problem into 7 subproblems of size $n/3$, each is solved in linear time. We can derive an equation as
   $T(n) = 7T(n/3) + \Theta(n)$

   The Master's theorem can be applied here. From the given equation, we can derive that
   $a = 7, b = 3, f(n) = \Theta(n)$

   Let,
   $g(n) = n^{\log_b a}$
   $g(n) = n^{\log_3 7}$
   $g(n) = n^{1.77}$

   We can see that,
   $f(n) \leq g(n)$
   $f(n) = \Theta(n) \leq n^{\log_3 7 - \epsilon}$
   $f(n) = \Theta(n) \leq n^{1.77 - \epsilon}$

   Therefore,
   $T(n) = \Theta(n^{\log_3 7}) = n^{1.77}$

   (b) (3 points) Algorithm B, divides the original problem into 4 subproblems of size $5n/7$, recursively solves the subproblems, and then solves the original problem by combining the subproblems in linear time. Provide recurrence formula and running time in Θ notation.

   Ans. B divides the problem into 4 subproblems of size $5n/7$, each is solved in linear time. We can derive an equation as

$T(n) = 4T(5n/7) + \Theta(n)$

The Master's theorem can be applied here. From the given equation, we can derive that

$a = 4, b = 7/5, f(n) = \Theta(n)$

Let,

$g(n) = n^{\log_b a}$

$g(n) = n^{\log_{7/5} 4}$

$g(n) = n^{4.12}$

We can see that,

$f(n) \leq g(n)$

$f(n) = \Theta(n) \leq n^{\log_{7/5} 4 - \epsilon}$

$f(n) = \Theta(n) \leq n^{4.12 - \epsilon}$

Therefore,

$T(n) = \Theta(n^{\log_{7/5} 4}) = n^{4.12}$

(c) (3 points) Algorithm C, divides the original problem into 4 subproblems of size $n/3$, recursively solves the subproblems, and then solves the original problem by combining the subproblems in $\Theta(n^2)$ time. Provide recurrence formula and running time in Θ notation.

Ans. C divides the problem into 4 subproblems of size $n/3$, each is solved in $\Theta(n^2)$ time. We can derive an equation as

$T(n) = 4T(n/3) + \Theta(n^2)$

The Master's theorem can be applied here. From the given equation, we can derive that

$a = 4, b = 3, f(n) = \Theta(n^2)$

Let,

$g(n) = n^{\log_b a}$

$g(n) = n^{\log_3 4}$

$g(n) = n^{1.26}$

We can see that,

$f(n) \geq g(n)$

$f(n) = \Theta(n^2) \geq n^{\log_3 4}$

$f(n) = \Theta(n^2) \geq n^{1.26}$

Therefore,

$T(n) = \Theta(n^2)$

(d) (1 point) Which of the above 3 is the fastest algorithm?

Ans. Algorithm A is the fastest.

$A \leq C \leq B$
$\Theta(n^{\log_3 7}) \leq \Theta(n^2) \leq \Theta(n^{log_{7/5}4})$

2. (10 points) Use the substitution method to prove that $T(n) = 2T(n/2)+cn\log n$ is $O(n\log^2 n)$.

Ans. We have to prove that $T(n) <= kn\log^2 n$

n/2 lies in between [0,n]

So, this can hold our assumption

$==> T(n/2) <= k(n/2)\log^2(n/2)$

$T(n) = 2T(n/2) + cn\log n$
$T(n) <= 2(k(n/2)\log^2(n/2)) + cn\log n$
$T(n) <= kn\log^2(n/2) + cn\log n$
$T(n) <= kn(\log^2 n + 1 - 2\log n) + cn\log n$
$T(n) <= kn\log^2 n + kn - 2kn\log n + cn\log n$
$T(n) <= kn\log^2 n + (c - 2k)n\log n + kn$
$T(n) <= kn\log^2 n + kn \quad \forall k > c/2$

Since the lower order terms can be ignored,
$T(n) = O(n\log^2 n)$

3. (10 points) Use the substitution method to prove that $T(n) = 4T(n/2) + n^2$ is $O(n^2\log n)$.

Ans.

We have to prove that $T(n) <= cn^2\log n$

n/2 lies in between [0,n]

So, this can hold our assumption

$==> T(n/2) <= c(n/2)^2\log(n/2)$

$T(n) = 4T(n/2) + n^2$
$T(n) <= 4(c(n/2)^2\log(n/2)) + n^2$
$T(n) <= 4((cn^2/4)\log(n/2)) + n^2$
$T(n) <= cn^2\log(n/2)) + n^2$
$T(n) <= cn^2(\log n - \log 2) + n^2$
$T(n) <= cn^2\log n - cn^2\log 2 + n^2$
$T(n) <= cn^2\log n + n^2(1 - c\log 2)$
$T(n) <= cn^2\log n \quad \forall c >= 1$

$T(n) = O(n^2\log n)$, Hence proved.

4. (15 points) Matrix multiplication:

(a) (5 points) Divide the 4 x 4 matrix A into 4 smaller matrices of size 2 x 2

$A = \begin{bmatrix} 1 & 2 & 3 & 4 \\ 5 & 6 & 7 & 8 \\ 9 & 10 & 11 & 12 \\ 13 & 14 & 15 & 16 \end{bmatrix}$ to create $\begin{bmatrix} A11 & A12 \\ A21 & A22 \end{bmatrix}$

Show 2 x 2 matrices $A11, A12, A21, A22$.

Ans.

$A_{11} = \begin{bmatrix} 1 & 2 \\ 5 & 6 \end{bmatrix}$ $A_{12} = \begin{bmatrix} 3 & 4 \\ 7 & 8 \end{bmatrix}$ $A_{21} = \begin{bmatrix} 9 & 10 \\ 13 & 14 \end{bmatrix}$ $A_{22} = \begin{bmatrix} 11 & 12 \\ 15 & 16 \end{bmatrix}$

(b) (5 points) Perform some of the calculations needed to compute A x B using Strassen's algorithm:

$\begin{bmatrix} 1 & 2 & 3 & 4 \\ 5 & 6 & 7 & 8 \\ 9 & 10 & 11 & 12 \\ 13 & 14 & 15 & 16 \end{bmatrix} \cdot \begin{bmatrix} 17 & 18 & 19 & 20 \\ 21 & 22 & 23 & 24 \\ 25 & 26 & 27 & 28 \\ 29 & 30 & 31 & 32 \end{bmatrix} = C$

For the matrices given above compute:

- Compute $P1, P2, C12$.
- Compute $A11.B12$ and $A12.B22$.
- Check to see that $C12 = A11.B12 + A12.B22$.

Ans.

Matrices A, B needs to be divided into four 2 x 2 matrices

$A_{11} = \begin{bmatrix} 1 & 2 \\ 5 & 6 \end{bmatrix}$ $A_{12} = \begin{bmatrix} 3 & 4 \\ 7 & 8 \end{bmatrix}$ $A_{21} = \begin{bmatrix} 9 & 10 \\ 13 & 14 \end{bmatrix}$ $A_{22} = \begin{bmatrix} 11 & 12 \\ 15 & 16 \end{bmatrix}$

$B_{11} = \begin{bmatrix} 17 & 18 \\ 21 & 22 \end{bmatrix}$ $B_{12} = \begin{bmatrix} 19 & 20 \\ 23 & 24 \end{bmatrix}$ $B_{21} = \begin{bmatrix} 25 & 26 \\ 29 & 30 \end{bmatrix}$ $B_{22} = \begin{bmatrix} 27 & 28 \\ 31 & 32 \end{bmatrix}$

C = A x B

$C = \begin{bmatrix} C_{11} & C_{12} \\ C_{21} & C_{22} \end{bmatrix}$

$= \begin{bmatrix} A_{11} & A_{12} \\ A_{21} & A_{22} \end{bmatrix} \cdot \begin{bmatrix} B_{11} & B_{12} \\ B_{21} & B_{22} \end{bmatrix}$

$= \begin{bmatrix} (A_{11}.B_{11}) + (A_{12}.B_{21}) & (A_{11}.B_{12}) + (A_{12}.B_{22}) \\ (A_{21}.B_{11}) + (A_{22}.B_{21}) & (A_{21}.B_{12}) + (A_{22}.B_{22}) \end{bmatrix}$

$P_1 = A_{11}(B_{12} - B_{22}) = \begin{bmatrix} -24 & -24 \\ -88 & -88 \end{bmatrix}$

$P_2 = (A_{11} + A_{12})B_{22} = \begin{bmatrix} 294 & 304 \\ 758 & 784 \end{bmatrix}$

$C_{12} = P_1 + P_2 = \begin{bmatrix} 270 & 280 \\ 670 & 696 \end{bmatrix} \implies Eqn1$

$A_{11}.B_{12} + A_{12}.B_{22} = (\begin{bmatrix} 1 & 2 \\ 5 & 6 \end{bmatrix} \cdot \begin{bmatrix} 19 & 20 \\ 23 & 24 \end{bmatrix}) + (\begin{bmatrix} 3 & 4 \\ 7 & 8 \end{bmatrix} \cdot \begin{bmatrix} 27 & 28 \\ 31 & 32 \end{bmatrix})$

$= \begin{bmatrix} 65 & 68 \\ 233 & 244 \end{bmatrix} + \begin{bmatrix} 205 & 212 \\ 437 & 452 \end{bmatrix}$

$$= \begin{bmatrix} 270 & 280 \\ 670 & 696 \end{bmatrix} \implies Eqn2$$

From above equations $Eqn1 \;\; Eqn2$, we can conclude that $C_{12} = A_{11}.B_{12} + A_{12}.B_{22}$.

5. (10 points) Design an efficient algorithm to multiply a n x $3n$ matrix with a $3n$ x n matrix where you use Strassen's algorithm as a subroutine. Justify your run time. No points will be given for an inefficient algorithm.

Ans.

Assume matrix A is of order n x 3n and matrix B is of order 3n x n.

$$A = \begin{bmatrix} a_{11} & a_{12} & ... & a_{1n} & ... & a_{12n} & ... & a_{13n} \\ a_{21} & a_{22} & ... & a_{2n} & ... & a_{22n} & ... & a_{23n} \\ ... & ... & ... & ... & ... & ... & ... & ... \\ ... & ... & ... & ... & ... & ... & ... & ... \\ a_{n1} & a_{n2} & ... & a_{nn} & ... & a_{n2n} & ... & a_{n3n} \end{bmatrix} B = \begin{bmatrix} b_{11} & b_{12} & ... & b_{1n} \\ b_{21} & b_{22} & ... & b_{2n} \\ ... & ... & ... & ... \\ ... & ... & ... & ... \\ b_{n1} & b_{n2} & ... & b_{nn} \\ ... & ... & ... & ... \\ b_{2n1} & b_{2n2} & ... & b_{2nn} \\ ... & ... & ... & ... \\ b_{3n1} & b_{3n2} & ... & b_{3nn} \end{bmatrix}$$

Let's assume, C = A x B where C is order of n x n.

$$C = \begin{bmatrix} C_{11} & C_{12} & C_{13} & ... & C_{1n} \\ C_{21} & C_{22} & C_{23} & ... & C_{2n} \\ C_{31} & C_{32} & C_{33} & ... & C_{3n} \\ ... & ... & ... & ... & ... \\ ... & ... & ... & ... & ... \\ C_{n1} & C_{n2} & C_{n3} & ... & C_{nn} \end{bmatrix}$$

Here,

$C_{11} = a_{11}b_{11} + a_{12}b_{21} + ... + a_{1n}b_{n1} + ... + a_{12n}b_{2n1} + ... + a_{13n}b_{3n1} \implies Eqn.1$

To compute A x B, we have to break matrix A, B into three n x n matrices where

$$A_1 = \begin{bmatrix} a_{11} & a_{12} & ... & a_{1n} \\ a_{21} & a_{22} & ... & a_{2n} \\ ... & ... & ... & ... \\ ... & ... & ... & ... \\ a_{n1} & a_{n2} & ... & a_{nn} \end{bmatrix}, A_2 = \begin{bmatrix} a_{1n+1} & a_{1n+2} & ... & a_{12n} \\ a_{2n+1} & a_{2n+2} & ... & a_{22n} \\ ... & ... & ... & ... \\ ... & ... & ... & ... \\ a_{nn+1} & a_{nn+2} & ... & a_{n2n} \end{bmatrix}, A_3 = \begin{bmatrix} a_{12n+1} & a_{12n+2} & ... & a_{13n} \\ a_{22n+1} & a_{22n+2} & ... & a_{23n} \\ ... & ... & ... & ... \\ ... & ... & ... & ... \\ a_{n2n+1} & a_{n2n+2} & ... & a_{n3n} \end{bmatrix}$$

$$B_1 = \begin{bmatrix} b_{11} & b_{12} & ... & b_{1n} \\ b_{21} & b_{22} & ... & b_{2n} \\ ... & ... & ... & ... \\ ... & ... & ... & ... \\ b_{n1} & b_{n2} & ... & b_{nn} \end{bmatrix}, B_2 = \begin{bmatrix} b_{n+11} & b_{n+12} & ... & b_{n+1n} \\ b_{n+21} & b_{n+22} & ... & b_{n+2n} \\ ... & ... & ... & ... \\ ... & ... & ... & ... \\ b_{2n1} & b_{2n2} & ... & b_{2nn} \end{bmatrix}, B_3 = \begin{bmatrix} b_{2n+11} & b_{2n+12} & ... & b_{2n+1n} \\ b_{2n+21} & b_{2n+22} & ... & b_{2n+2n} \\ ... & ... & ... & ... \\ ... & ... & ... & ... \\ b_{3n1} & b_{3n2} & ... & b_{3nn} \end{bmatrix}$$

and

$A = \begin{bmatrix} A_1 & A_2 & A_3 \end{bmatrix}$ and $B = \begin{bmatrix} B_1 \\ B_2 \\ B_3 \end{bmatrix}$

So, assume that

A x B = [(A_1 x B_1) + (A_2 x B_2) + (A_3 x B_3)]

Here,

$$A_1 \text{ x } B_1 = \begin{bmatrix} a_{11} & a_{12} & \dots & a_{1n} \\ a_{21} & a_{22} & \dots & a_{2n} \\ \dots & \dots & \dots & \dots \\ \dots & \dots & \dots & \dots \\ a_{n1} & a_{n2} & \dots & a_{nn} \end{bmatrix} \text{ x } \begin{bmatrix} b_{11} & b_{12} & \dots & b_{1n} \\ b_{21} & b_{22} & \dots & b_{2n} \\ \dots & \dots & \dots & \dots \\ \dots & \dots & \dots & \dots \\ b_{n1} & b_{n2} & \dots & b_{nn} \end{bmatrix}$$

$$A_2 \text{ x } B_2 = \begin{bmatrix} a_{1n+1} & a_{1n+2} & \dots & a_{12n} \\ a_{2n+1} & a_{2n+2} & \dots & a_{22n} \\ \dots & \dots & \dots & \dots \\ \dots & \dots & \dots & \dots \\ a_{nn+1} & a_{nn+2} & \dots & a_{n2n} \end{bmatrix} \text{ x } \begin{bmatrix} b_{n+11} & b_{n+12} & \dots & b_{n+1n} \\ b_{n+21} & b_{n+22} & \dots & b_{n+2n} \\ \dots & \dots & \dots & \dots \\ \dots & \dots & \dots & \dots \\ b_{2n1} & b_{2n2} & \dots & b_{2nn} \end{bmatrix}$$

$$A_3 \text{ x } B_3 = \begin{bmatrix} a_{12n+1} & a_{12n+2} & \dots & a_{13n} \\ a_{22n+1} & a_{22n+2} & \dots & a_{23n} \\ \dots & \dots & \dots & \dots \\ \dots & \dots & \dots & \dots \\ a_{n2n+1} & a_{n2n+2} & \dots & a_{n3n} \end{bmatrix} \text{ x } \begin{bmatrix} b_{2n+11} & b_{2n+12} & \dots & b_{2n+1n} \\ b_{2n+21} & b_{2n+22} & \dots & b_{2n+2n} \\ \dots & \dots & \dots & \dots \\ \dots & \dots & \dots & \dots \\ b_{3n1} & b_{3n2} & \dots & b_{3nn} \end{bmatrix}$$

and we have,

A x B = [(A_1 x B_1) + (A_2 x B_2) + (A_3 x B_3)]

and

$$A \text{ x } B = \begin{bmatrix} a_{11}b_{11} + a_{12}b_{21} + \dots + a_{1n}b_{n1} + \dots + a_{12n}b_{2n1} + \dots + a_{13n}b_{3n1} & \dots & \dots \\ \dots & \dots & \dots \\ \dots & \dots & \dots \end{bmatrix}$$

From $Eqn.1$, we have $C_{11} = a_{11}b_{11} + a_{12}b_{21} + \dots + a_{1n}b_{n1} + \dots + a_{12n}b_{2n1} + \dots + a_{13n}b_{3n1}$

$$A \text{ x } B = \begin{bmatrix} C_{11} & \dots & \dots \\ \dots & \dots & \dots \\ \dots & \dots & \dots \end{bmatrix}$$

Similarly, we can prove this for other terms like $C_{12}, C_{13}, \dots, C_{nn}$ and we will have

$$A \text{ x } B = \begin{bmatrix} C_{11} & C_{12} & \dots & C_{1n} \\ C_{21} & C_{22} & \dots & C_{2n} \\ \dots & \dots & \dots & \\ C_{n1} & C_{n2} & \dots & C_{nn} \end{bmatrix}$$

To compute A_1 x B_1, A_2 x B_2, A_3 x B_3; we can call Strassen's algorithm as a subroutine.

Here, it takes **3 matrix multiplications of n x n order and 2 matrix additions of n x n order**.

$T(n) = 3(7T(n/2) + \Theta(n^2)) + 2n^2$

The above recursive equation can be solved using the Master's Theorem and the time complexity is $\Theta(n^{\log_2 7})$.