(1)

الاسم : هازم عادل محمد أحمد

ID : 20210269

Algorithms - Hw#2

- Algorithm 1

a) Assume: $A \begin{bmatrix} 1 & 2 \\ 3 & 4 \end{bmatrix}$, $B \begin{bmatrix} 1 & 2 \\ 3 & 4 \end{bmatrix}$

$C \begin{bmatrix} 0 & 0 \\ 0 & 0 \end{bmatrix}$, $n = 2$

output: $C \begin{bmatrix} 1 & 0 \\ 0 & 0 \end{bmatrix}$ at $i, j, k = 0$, $C \begin{bmatrix} 2 & 0 \\ 0 & 0 \end{bmatrix}$ at $k = 1$, i, j

$C \begin{bmatrix} 2 & 4 \\ 0 & 0 \end{bmatrix}$ at $i=0$, $j=1$, $k=0$, $C \begin{bmatrix} 2 & 8 \\ 0 & 0 \end{bmatrix}$ $i=0$, $j=1$, $k=1$

$C \begin{bmatrix} 2 & 4 \\ 9 & 0 \end{bmatrix}$ at $i=1$, $j=0$, $k=0$, $C \begin{bmatrix} 2 & 4 \\ 18 & 0 \end{bmatrix}$ at $i=1$, $j=0$, $k=1$

$C \begin{bmatrix} 2 & 4 \\ 18 & 16 \end{bmatrix}$ at $i=1$, $j=1$, $k=0$, $C \begin{bmatrix} 2 & 4 \\ 18 & 32 \end{bmatrix}$ $i=1$, $j=1$, $k=1$

~~Basic~~

B) Basic operation: $C[I,J] \leftarrow C[I,J] + A[I,J] * B[$

c) $\sum_{i=0}^{n-1} \sum_{j=0}^{n-1} \sum_{k=0}^{n-1} 1 = n^3 = \Theta(n^3)$

طارق عماد حسين أحمد

(2) 20210269

- Algorithm 2

a) Assume: $n = 3$

output: $S = 0+1+1 = 2$ at $i=1$

$S = 2+2+2 = 6$ at $i = 2$

$S = 6+3+3 = 12$ at $i = 3$

return $\longrightarrow$ (12)

b) Basic operation: $S \leftarrow S + i + i$

c) $C(n) = \sum_{i=1}^{n} 1 = n - 1 + 1 = n \longrightarrow \Theta(n)$

d) no worst case, because it's gonna always take n times

- Algorithm. 3

a) Assume: $A[1, 3, 5]$, $n = 3$

output: $r = 1$, $S = 5$, return: $5 - 1 = 4$

b) Basic operation: if $A[i] < r$, if $A[i] > S$

c) $C(n) = \sum_{i=1}^{n-1} n - 1 = \cancel{1+1} = n - 1 \longrightarrow \Theta(n)$

d) all of cases will take n times

20210269 حازم عاد حسين أحمد

(3)

- Algorithm 4

a) Assume : A $\begin{bmatrix} 1 & 2 \\ 3 & 4 \end{bmatrix}$

output: return → False $1 \neq 3$

b) Basic operation : $\neq$

c) $C(n) = \sum_{i=1}^{n-2} \sum_{j=i+1}^{n-1} 1 = \sum_{i=1}^{n-2} (n-1) \cancel{-(i+1)+1} = n-1-i$

$= \sum_{i=1}^{n-2} n-1-i = \sum_{i=1}^{n-2} n-1 - \sum_{i=1}^{n-2} i = (n-1)\sum_{i=1}^{n-3} 1 - \sum_{i=1}^{n-2} i$

$= (n-1) \cdot (n-2) - \frac{(n-1) \cdot (n-2)}{2} \longrightarrow \Theta(n^2)$

d) Best case $\Omega(1)$, Worst case $O(n^2)$

2021 0269

طاهر عادل حسن أحمد

(4)

2- A = < E, X, A, M, P, L, E >

E | X A M P L E

E X | A M P L E

A E X | M P L E

A E M X | P L E

A E M P X | L E

A E L M P X | E

A E E L M P X → A < A, E, E, L, M, P, X >

3- A = < 3, 41, 52, 26, 38, 57, 9, 49 >

3 | 41, 52, 25, 38, 57, 9, 49

3 41 | 52, 25, 38, 57, 9, 49

3 41 52 | 25 38 57 9 49

3 25 41 52 | 38 57 9 49

3 25 38 41 52 | 57 9 49

3 25 38 41 52 57 | 9 49

3 9 25 38 41 52 57 | 49

3 9 25 38 41 49 52 57

A = < 3, 9, 25, 38, 41, 49, 52, 57 >

20210269

مازن عادل محمد أحمد

(5)

4. Consider the following version

$C(n) = \sum_{i=1}^{n-1} 1 = n-1-1+1 = n-1 \rightarrow \Theta(n)$

worst case is $O(n^2)$ if it's reversed

- The difference between the 2 versions is that this version uses swap not shifting

5- insertion sort best case $\rightarrow \Theta(n)$
Bubble sort best case $\rightarrow \Theta(n^2)$
- so insertion sort is better.

6- both of them has the same time complexity $\Theta(n^2)$