Name : Sradha Kedia
Date of Examination : 16 December, 2021
Time of Examination : 9:00 am to 1:00 pm
Examination Roll no. : 20234757053
Semester : III
Unique Paper Code : 22340301
Title of Paper : Design and Analysis of Algorithm
Email-Id : 200083@cs.du.ac.in
Mobile no. of student : ~~200083@cs.du.ac.in~~ 9899519848
Question no. : 1
No. of pages : 7
Name of the program : MCA
Name of the Department : DUCS

2(b)

Jobs:	J_1	J_2	J_3	J_4	J_5	J_6
Deadlines:	5	3	3	2	4	2
Profits:	200	180	190	300	120	100

no. of jobs = 6
max deadline = 5
∴ no. of time slots available are 5

① Sort the jobs on the basis of profits

Jobs →	J_4	J_1	J_3	J_2	J_5	J_6
Deadline →	2	5	3	3	4	2
Profit →	300	200	190	180	120	100

② Available slots are

$$0 \xrightarrow{J_2} 1 \xrightarrow{J_4} 2 \xrightarrow{J_3} 3 \xrightarrow{J_5} 4 \xrightarrow{J_1} 5$$

Allocating slots on basis of optimal schedule i.e. selecting jobs with maximum profit first.

Job consider	slot assign	sequence of Jobs	Profit
J_4	$1 \rightarrow 2$	J_4	300
J_1	$4 \rightarrow 5$	J_4, J_1	500
J_3	$2 \rightarrow 3$	J_4, J_1, J_3	690
J_2	$0 \rightarrow 1$	J_4, J_1, J_3, J_2	870
J_5	$3 \rightarrow 4$	J_4, J_1, J_3, J_2, J_5	990

∴ optimal schedule that gives optimal profit is:
$J_2 \rightarrow J_4 \rightarrow J_3 \rightarrow J_5 \rightarrow J_1$

no, all the jobs are not completed in the optimal schedule as J_6 is not completed.

max profit $= P(J_2) + P(J_4) + P(J_3) + P(J_5) + P(J_1)$
$= 180 + 300 + 190 + 120 + 200$
$= 990$

∴ max profit earned is 990.

2 (c) Algo to find max & min element in the array
Input : integer array ⇒ int arr []
: int arr []
: size of array - int n
output : Prints the min & max element
Return : void
Assumption : None

Algorithm:

```
void findMinMax(int arr[], int n)
{
    int min = arr[0];
    int max = arr[0];
    for (int i = 1; i < n; i++) {
        if (arr[i] > max)
            max = arr[i];
        else if (arr[i] < min)
            min = arr[i];
    }
    print max ele;
    print min ele;
```

Analysis:

Best case → when elements are sorted in ascending order in this situation only first comparison is made for $n-1$ times.

worst case → when elements are sorted in descending order in this case first condition will be false and second comparison will always will be there.

∴ Almost $2(n-1)$ comparisons will be there.

Time Complexity : $O(n)$

space " : $O(1)$

Also, there is an algo that we improve by comparing pairs of elements from the input first with each other and then the compare minimum with current minimum and maximum with current

maximum at a cost of 3 comparisons for every two elements.

intial values for current max. & current min depends on whether n is odd or even.
→ if n is odd, no. of comparisons is

$$\frac{3(n-1)}{2}$$

→ if n is even, $\frac{3n}{2} - 2$.

so, it is optimised approach than the first written but acc. to condition given in ques the algo written spicify the case.

(a) Multiply (a, b):
① assume n = length(a) = length(b)
② if length(a) <= 1 then return a*b
③ Partition a, b into

$$a = a_1 * 10^{n/2} + a_2$$
$$b = b_1 * 10^{n/2} + b_2$$

④ find

$$A = \text{multiply}(a_1, b_1)$$
$$B = \text{multiply}(a_2, b_2)$$
$$C = \text{multiply}(a_1 + a_2, b_1 + b_2)$$

⑤ Return $A * 10^n + (C - A - B) \times 10^{n/2} + B$

Thus, $T(n) = 3\,T(n/2) + O(n)$

$$\sum_{i=0}^{\log_2 n} 3^i A \frac{n}{2^i}$$

$= O(n \frac{3}{2} \log n)$

$= O(n \times n \log 2^{3/2})$

$= O[n \log_2 3]$ $O(n^{\log_2 3}) = O(n)$

123456 × 654321

$x = 123 | 456$

$y = 654 | 321$

① n = length of $x = 6$

② $x = 123 \times 10^3 + 456$

$y = 654 \times 10^3 + 321$

③ multiply (123, 654)

$x_1 = 01 \times 10^2 + 23$

$y_1 = 06 \times 10^2 + 54$

from step ② of above algo

$A_1 = 6 = 01 \times 06$

$B_1 =$ Multiply (23, 54)

$x_2 = 2 \times 10 + 3$

$y_2 = 5 \times 10 + 4$

$23 \times 54 = 10 \times 10^2 + (45 - 10 - 12) \times 10 + 12$

$= 10^3 + 230 + 12$

$= 1242$

so, $B_1 = 1242 = 023 \times 054$

$C_1 =$ Multiply (24, 60)

$x_3 = 2 \times 10 + 4$

$y_3 = 6 \times 10 + 0$

$A_3 = 12$

$B_3 = 0$

$C_3 = 48$

$x_3 \times y_3 = 12 \times 10^2 + (48 - 12) \times 10 + 0$

$C_1 \Rightarrow 1440$

Now, $x_1 \times y_1 = 6 \times 10^4 + (1440 - 6 - 1242) \times 10^2 + 1242$

$= 80442$

$123 \times 654 = 80442$

now, $B =$ multiply $(456, 321)$

$x_1 \times y_1 = (04 \times 10^2 + 56) \times (03 \times 10^2 + 21)$

$A_1 = 12$

$B_1 =$ multiply $(56, 21)$

$x_2 = 5 \times 10 + 6$

$y_2 = 2 \times 10 + 1$

$56 \times 21 = 12 \times 10^2 + (33 - 12 - 6) \times 10 + 6$

$B_1 = 1176 = 56 \times 21$

$C_1 =$ multiply $(60, 24)$

$x_3 = 6 \times 10 + 0$

$x_4 = 2 \times 10 + 4$

$60 \times 24 = 1440$

$B = 12 \times 10^4 + (1440 - 1176 - 12) \times 10^2 + 1176$

$\boxed{B = 146376}$

C = multiply (579, 975)

Ifly, we get

$\boxed{C = 564525}$

Step 5 → 123456×654321

$= 80442 \times 10^6 + (564525 - 80442 - 146376) \times 10^3 + 146376$

$= 80779853376$