Marwadi Education Foundation's Group of Institutions

Answer Book

Enrolment No.

9	2	2	0	0	1	3	3	0	3	0	

Aryandi
Student's Signature

Supervisor's Signature

Supervisor's Name: NA

Seat No.

MB-16

Affix the Barcode Sticker Here after reading the Instructions for Candidates on Page-2

Main	Suppli.	Total
1	+ 2 =	3

Subject Name	DAA
Subject Code	0 1 C T 0 5 1 2 - -
Date of Exam	2 7 - 0 9 - 2 0 2 4
Semester	5th
Course Name	B. Tech - ICT

PART - I (To be Filled by Examiner only)

Question No.	Marks by Examiner	Marks by Verifier-I	Marks by Verifier-II	For the use of University
1	8.75			
2	3.5		31.75 = 32/50	
3	4.0			
4	12.5			
5				
6				
7				
Total Marks Obtained				Signature of Supervisor (Incase of student registered under unfair Means)
Total Marks rounded				Signature of Sr.Supervisor (Incase of student registered under unfair Means)
Total Marks in words				

Jr. Supervisor Information

Name : NA	Date : 27/9	Sign :

Examiner Information

Name :	Date :	Sign :

IMPORTANT NOTE : Examiner is required to enter marks in LED form.

Que. 1	Que. 2	Que. 3	Que. 4	Que. 5	Que. 6	Que. 7	Total

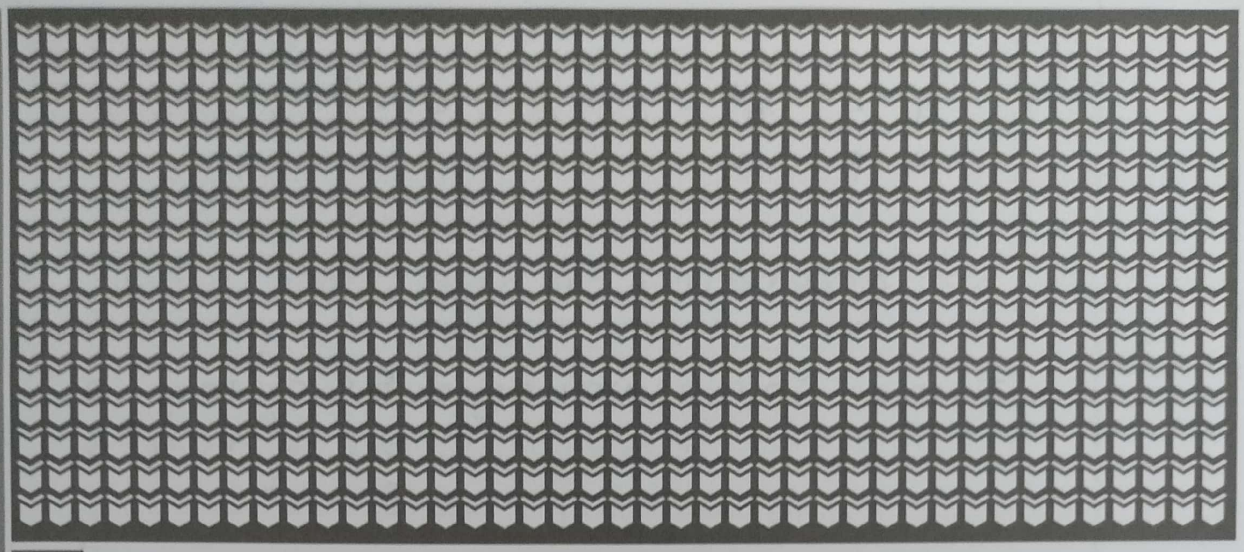
DO NOT WRITE ANYTHING IN THIS PORTION

Marwadi education foundation
Marwadi UNIVERSITY

Marwadi Education Foundation's Group of Institutions
Rajkot-Morbi Highway, Gauridad, Taluka & Dist. Rajkot - 360 003. Gujarat - India. Tel. (0281) 2924155 / 56
Mobile : 97277 24688 / 97277 24689 | Email : info@marwadieducation.edu.in | www.marwadieducation.edu.in

Scanned with OKEN Scanner

DO NOT WRITE ANYTHING IN THIS PORTION



Q. No.

Question - 1

Ans - 1

For a Sorted array, Binary Search is the most optimized algo. having Time Complexity O(log n)

Ans - 2

The tight bound complexity :-

$\theta(n^7)$ $\Omega(n^7)$ 0.5

Ans - 3

Recurrence Relation :-

$T(n) = 3T(n/5) + n^2$

$a = 3, \quad b = 5, \quad k = 2, \quad p = 0$

Scanned with OKEN Scanner

Q. No.

$a < b^k$ (i.e. $3 < 5^2$)

and $p = 0$

so according to the case - 3a

Time complexity $\Rightarrow \Theta(n^k \log^p n)$

$\Theta(n^2 \log^0 n)$

$\Theta(n^2)$

Ans - 4

→ For inserting a node at the beginning of Linked List will be $O(1)$ (1)

Ans - 5

The time complexity will be $\Theta(\min(k(N-k)))$ $O(n)$ (1)

Ans - 6

→ If input size doubles then the TC will remain same which is $O(n^2)$ (1)

Scanned with OKEN Scanner

DO NOT WRITE ANYTHING IN THIS PORTION

Q. No.

Ans - 7

1

Time complexity to insert the already a element in the array O(n-k) ≈ O(n)

where n → input size
k → position

because we have to move each element one position right side after k^th position.

Ans - 8

0.75

Array → O(k) O(1)
LinkedList → O(k)
stack → O(k)
Queue → O(k)

Ans - 9

newPage → O(1)
backward → O(1)
forward → O(1)

1

Ans - 10

enque → O(n) O(1)
deque → O(n)

0.5

Scanned with OKEN Scanner

Q. No.

Question - 2

Ans - A

outer loop :- it will run $n/2 - n/4 = n/2$ times

inner loop :- it will run $n - n/4 - 1 - 3n - 4/4$ times

Total operations :- $\frac{n/2 \times 3n-4}{4}$

$\therefore \frac{n(3n-4)}{8}$

$\therefore \frac{3n^2 - 4n}{8}$

$F(n) = \frac{3}{8} n^2 - \frac{n}{2}$

so, the complexity will be :-
$O(n^2)$

Upper bound :- $c_1 . g(n) > F(n)$

$\frac{4}{8} n^2 > \frac{3}{8} n^2 - \frac{n}{2}$

$\frac{1}{8} n^2 > -\frac{n}{2}$

$\frac{1}{8} n^2 + \frac{n}{2} > 0$

Scanned with OKEN Scanner

Q No.

$\frac{1}{8}n^2 + \frac{n}{2} > 0$

$2n^2 + 8n > 16$

$2n^2 + 8n - 16 > 0$

$n^2 + 4n - 8 > 0$

$b^2 - 4ac \Rightarrow 16 - 4(1)(-8)$

$16 + 32$

48

$n = \frac{-b + \sqrt{48}}{2}$

Here $\sqrt{48} \approx 6 < 2(n)$

So, the value of n cannot be positive

We can't find upper bound

Ans - B

DO NOT WRITE ANYTHING IN THIS PORTION

Scanned with OKEN Scanner

8 ☐ + ☐ + ☐ + ☐ + ☐ + ☐ = ☐

Q. No.

Question - 2

Ans- A

$Fcn = (n/2 - n/4) \times \frac{3n-4}{4} \times \log_6 n$

1.5

$= n^2 \log n$

because outer loop of i will run $n/2 - n/4 = n/2$ times

j loop will run $n - n/4 - 1$ times $= 3n-4/4$ times

k loop will run $\log n$ times

So, worst case time complexity $O(n^2 \log n)$

$Fcn = \frac{n}{2} \times \frac{3n-4}{4} \times \log n$

$= \frac{(3n^2 - 4n)}{8} \log n$

$= \frac{3n^2 \log n - 4n \log n}{8}$

$= \frac{3}{8} n^2 \log n - \frac{n}{2} \log n$

DO NOT WRITE ANYTHING IN THIS PORTION

Scanned with OKEN Scanner

Q. No.

For upper bound

$c_1 g(n) > f(n)$

$\frac{4}{8} n^2 \log n > \frac{3}{8} n^2 \log n - \frac{n \log n}{2}$

$\frac{1}{8} n^2 \log n > -\frac{n \log n}{2}$

$n \log n > -\log n$

$n \log n + \log n > 0$

$(n+1) \log n > 0$

this condition will satisfy for $\forall n > 1$ with $c_1 = 4$

Ans - 2

~~For upper~~

Here $f(n) =$ ~~n~~ n

For upper bound

$c_1 g(n) > f(n)$

$2n > n$

$2n - n > 0$

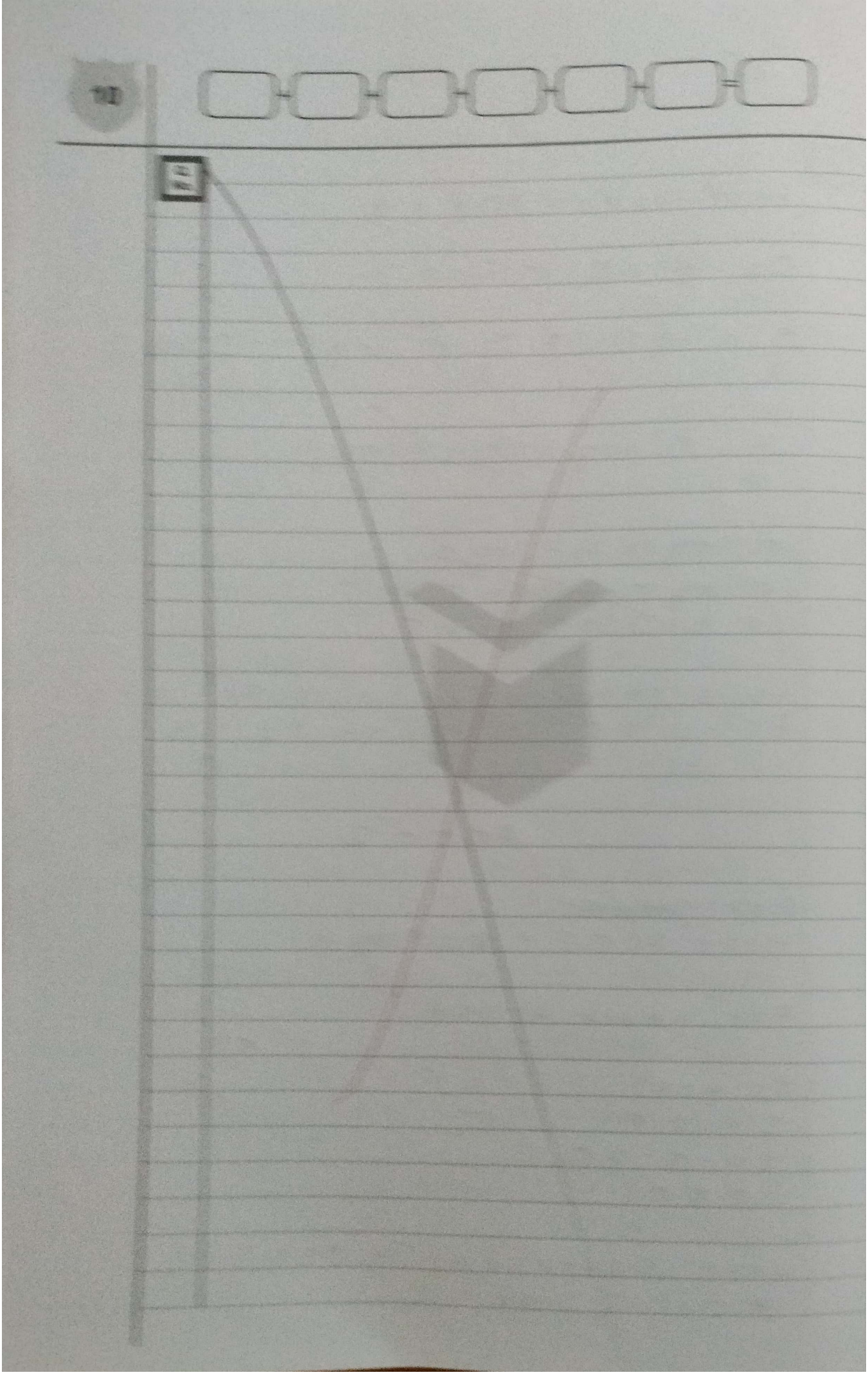
$n > 0$

Here upper bound $\rightarrow O(n)$

$c_1 = 2$, $n_0 = 1$

DO NOT WRITE ANYTHING IN THIS PORTION

Scanned with OKEN Scanner



Scanned with OKEN Scanner

DO NOT WRITE ANYTHING IN THIS PORTION

DO NOT WRITE ANYTHING IN THIS PORTION

Q. No.

Question - 3

Ans - 1

$T(n) = 10\,T(n/3) + 2n^{1.5}$

$a = 10$ $\quad k = 1.5$

$b = 3$ $\quad p = 0$

$a > b^k \;\because\; (10 > 3^{1.5})$ (1.5)

So, case - 1 (2)

then $T(n) = \theta(n^{\log_b a})$

$= \theta(n^{\log_3 10})$

$= \theta(n^{1.5})$

Ans - 2

$T(n) = 7T(n/4) + n^2 \log n$

$a = 7$ $\quad k = 2$

$b = 4$ $\quad p = 1$ (2)

$a < b^k \;\because\; (7 < 4^2)$

according to case 3 - a

$T(n) = \theta(n^k \log^p n)$

$= \theta(n^2 \log n)$

Scanned with OKEN Scanner

Q No.

Question - 4

Ans - 2

$f(n) = 3\sqrt{n} + 59$

For upper bound

$c \cdot g(n) > f(n)$

For $n \geq n_0$

so, For $c = 4$

$4\sqrt{n} > 3\sqrt{n} + 59$

$\sqrt{n} > 59$

$n > 3481$

Here $c = 4$ and $n = 3481$

$O(\sqrt{n})$

1.5

Ans - 3

$f(n) = 196n^2 + 35$

upper bound

2

$c_2 g(n) > f(n)$

$197n^2 > 196n^2 + 35$

$n^2 > 35$

$n > \sqrt{35}$ $\quad \because \sqrt{35} \approx 5.91$

$n > 5$

$c_1 = 197, \quad n_0 = 5$

DO NOT WRITE ANYTHING IN THIS PORTION

Scanned with OKEN Scanner

Q. No.

lower bound

$c_2 g(n) < f(n)$

$195 n^2 < 196 n^2 + 35$

$-35 < n^2$

$n^2 > -35$

For any $n > 0$ condition will satisfy

So, $c_2 = 195$, $n = 1$

Here $195 n^2 \leq 196 n^2 + 35 \leq 197 n^2$

So, Worst case $\Rightarrow O(n^2)$

Best case $\Rightarrow O(n^2)$

So, the average case will be $\theta(n^2)$

Hence proved that,

$f(n) = 196 n^2 + 35 \neq \Omega(n^3)$

$\because \Omega(n^2) \neq \Omega(n^3)$

Scanned with OKEN Scanner

Q. No.

Ans - 4

$F(n) = 1/2\,(3n^2 - 2n)$

$= 3/2\,n^2 - n$

Upper bound

$c_1 g(n) > F(n)$

$2n^2 > 3/2\,n^2 - n$

$1/2\,n^2 > -n$

$1/2\,n^2 + n > 0$

$n^2 + 2n > 0$

$n(n+2) > 0$

$n = 0$

Here $c_1 = 2$, $n_0 = 1$, $g(n) = n^2$

lower bound

$c_2 g(n) < F(n)$

$n^2 < 3/2\,n^2 - n$

$n < 3/2\,n^2 - n^2$

$1/2\,n^2 > n$

$n^2 > 2n$

$n^2 - 2n > 0$

$n(n-2) > 0$

$n > 0$ or $n > 2$

So $c_2 = 1$, $n_0 = 2$, $g(n) = n^2$

4

DO NOT WRITE ANYTHING IN THIS PORTION

DO NOT WRITE ANYTHING IN THIS PORTION

Scanned with OKEN Scanner

Marwadi University

Supplementary

Course Name	
Subject Name	
Date of Exam	

Enrollment No.

Supplementary No.

Student's Signature

Supervisor's Signature

Supervisor's Name

Seat No.

MU

EXAMINER HAS TO COMPULSORY ENTER MARKS IN BOX BELOW ON EACH PAGE

+ + + + + =

1

DO NOT WRITE ANYTHING IN THIS PORTION

Q. No.

Hence, $O(n^3)$

Here for $f(n) = 4n^3 + 2n$

O bound will be $O(n^3)$ with

$c_1 = 8$, $c_2 = 1$, $n_0 = 2$, $g(n) = n^3$

Ans = 7

```
def helper(max_wt, food_item):
    for food in food_items:
        if food.
```

2 + + + + + =

Q. No.

1

Ans - 5

Distance (P1, P2) → ??

~~return sqrt((P1.x - P2.x)^2~~

return sqrt(pow(P1.x - P2.x, 2)
    + pow(P1.y - P2.y, 2))

(x, y) ??

Find_Max (Arr)
~~for i in Arr~~
max = Arr[0]
for i in range(1, len(Arr)):
    if Arr[i] > max
        max = Arr[i]

O(n)

return max

~~Fi~~

Divide & Conquer ??

DO NOT WRITE ANYTHING IN THIS PORTION

DO NOT WRITE ANYTHING IN THIS PORTION

Scanned with OKEN Scanner

Q. No.

Divide (Points [ ])

{

left = _

DO NOT WRITE ANYTHING IN THIS PORTION

Scanned with OKEN Scanner

4 ☐ + ☐ + ☐ + ☐ + ☐ + ☐ = ☐

Q. No.

Ans - 7

~~Select (Food cfood i~~
~~main~~

4

food_items = {'Milk', 65, 4},
{'Butter', 740, 82},
{'cheese', 310, 25},
{'Ice cream', 170, 7}, {'Roasted
Peanuts', 570, 49},
{'Choclate Biscuit', 520, 28},
{'cream', 200, 21}

max_fat = 15

max_cal =

Select Food (food_items, max_
fat)

How??

DO NOT WRITE ANYTHING IN THIS PORTION

Scanned with OKEN Scanner

DO NOT WRITE ANYTHING IN THIS PORTION

Marwadi University

Supplementary

Course Name	

Subject Name	
Date of Exam	

Enrolment No.

Student's Signature

Supplementary No.: 3

Supervisor's Signature

Supervisor's Name	
Seat No.	
MU	

EXAMINER HAS TO COMPULSORY ENTER MARKS IN BOX BELOW ON EACH PAGE

+ + + + + = 1

Q. No.

Ans - 7

```
def select_food(max_wt, foods):
    cal = 0
    foods[0].calories
    fat = 0
    foods[0].fat
    curr_wt = 100

    for food in foods:
        if(foods[0].calories
```

DO NOT WRITE ANYTHING IN THIS PORTION

DO NOT WRITE ANYTHING IN THIS PORTION

```
Select_Food (Food_item max_Cal):
    Sorted_Food = Sort the Food items
                  in decreasing order
                  of Calories/Cal
    max_cal = 0
    For Food in Sorted_Foods
        if
        max_Cal -= Food[0]
        max_cal += Food[1]
    return max_cal

main()
max_Cal = 15
Food_item = [[ ]]
```

Scanned with OKEN Scanner

Q. No.

170
516

Rup work 6

sort - func()

selectFood (food_items, max_fat)
sort (food_items . begin(),
food_items . end(), sort_fun

max_fat = 150 123 cal = 170

	cal	fat
Ice cream	170	7
choc bist	520	28
milk	65	4
cheese	310	25
Rosted	570	49
cream	210	21
butter	204	
	740	82

max_fat = 151 , max_cal = 0
144 170
116 686

DO NOT WRITE ANYTHING IN THIS PORTION

Scanned with OKEN Scanner

4 ▢ + ▢ + ▢ + ▢ + ▢ + ▢ = ▢

Q. No.

Rub work

1 , 2 , 3 , 4 , 5

↑

3 n/2

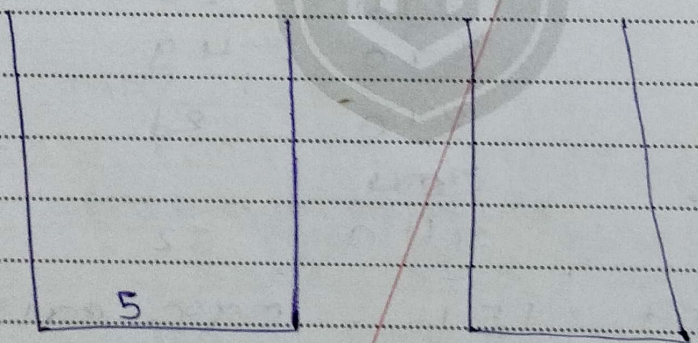
5 - 3 + 1 = 3

For (i = n+1; i >= n-k+1; i--) {

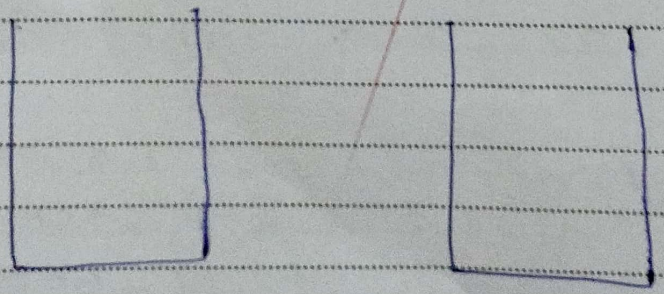
A[i] = A[i-1];

A[i] = key

1 , 2 , 3 , 4 , 5



1 2 3 4 5 OC



DO NOT WRITE ANYTHING IN THIS PORTION

Scanned with OKEN Scanner