

Expected Output:

En: The

principles later

• 1
• 2
• 3

Enter number of customer : 3100-90-65

Enter 3 Customer ID's:

4654686

1345484

2546561

~~front~~ : ~~front~~

Enter the Customer ID to search: 2546561

Middle term = 1^{st} , x^{th} oscillation : e.g. 2

2546561 found at location 2

ti ordei, così io riuscii a vederne i capelli.

Sample Outputs for 5 words

mettong' est bnd 926, o= [

Enter number of customers: 8

Enter gas customer ID's: bost : & got

789 + 943 = 1732 i start and it is in 1st column

4654686 ~~water~~-feed bar a-Fractionab-feed

13454841999999 - from kro "ref" ticket

9879876

90/2- : 29/2

2546461

3454848

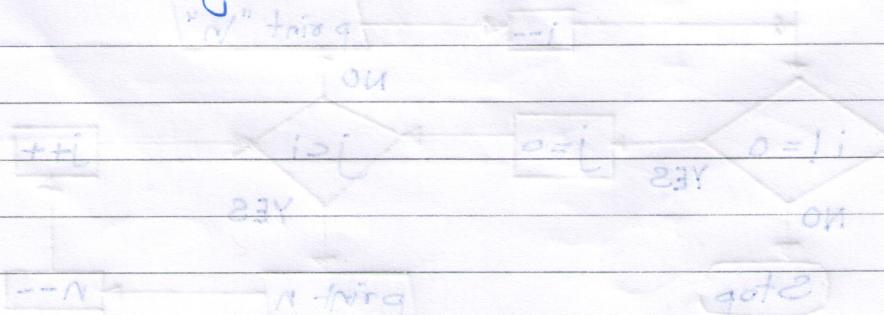
; food and }

6654862

Enter the customer ID to search: 8894654

Middle term = 3

8894654 found at location 7.



Ex No.: 5 Title:

Date:

04-10-2021

1 D Arrays

Algorithm:

Step 1: Start

Step 2: Declare i, j, fst, lft, mid, n, search, cid [100] and t.

Step 3: Read number of customers and Customer IDs.

Step 4: Introduce an outer for-loop, where i=0, it checks if i is lesser than n. If yes, let j = i + 1, else, exit the loop.

Step 5: Introduce an inner for-loop, where it checks if j is less than n. If yes, check if cid [j] < cid [i] and post-increment j, else, post-increment i and repeat the outer for-loop.

Step 6: If cid [j] < cid [i], then t = cid [i], cid [:] = cid [:j] and cid [j] = t.
Else, exit the loop.

Step 7: Read Customer ID to be searched and let fst = 0, lft = n - 1 and mid = (fst + lft) / 2.

Step 8 : Introduce a while loop that continues if $fst \leq lst$. In the loop, check if $cid[mid]$ is less than search. If yes, $fst = mid + 1$ and $mid = (fst + lst) / 2$, else, Check if $cid[mid]$ is equal to search. If yes, Display the position of the customer ID, else, $lst = mid - 1$ and $mid = (fst + lst) / 2$

Step 9 : Check if $fst > lst$. If yes, Display customer ID not found.

Flowchart :

