Ex. No: 5 **Date: 04-10-2021**

1D ARRAYS

PROBLEM GIVEN:

Write a program to store the data of customers and search for a given costumer by name or number and display details of customer.

ALGORITHM:

Step 1: Start

Step 2: Declare i, j, fst, lst, 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.

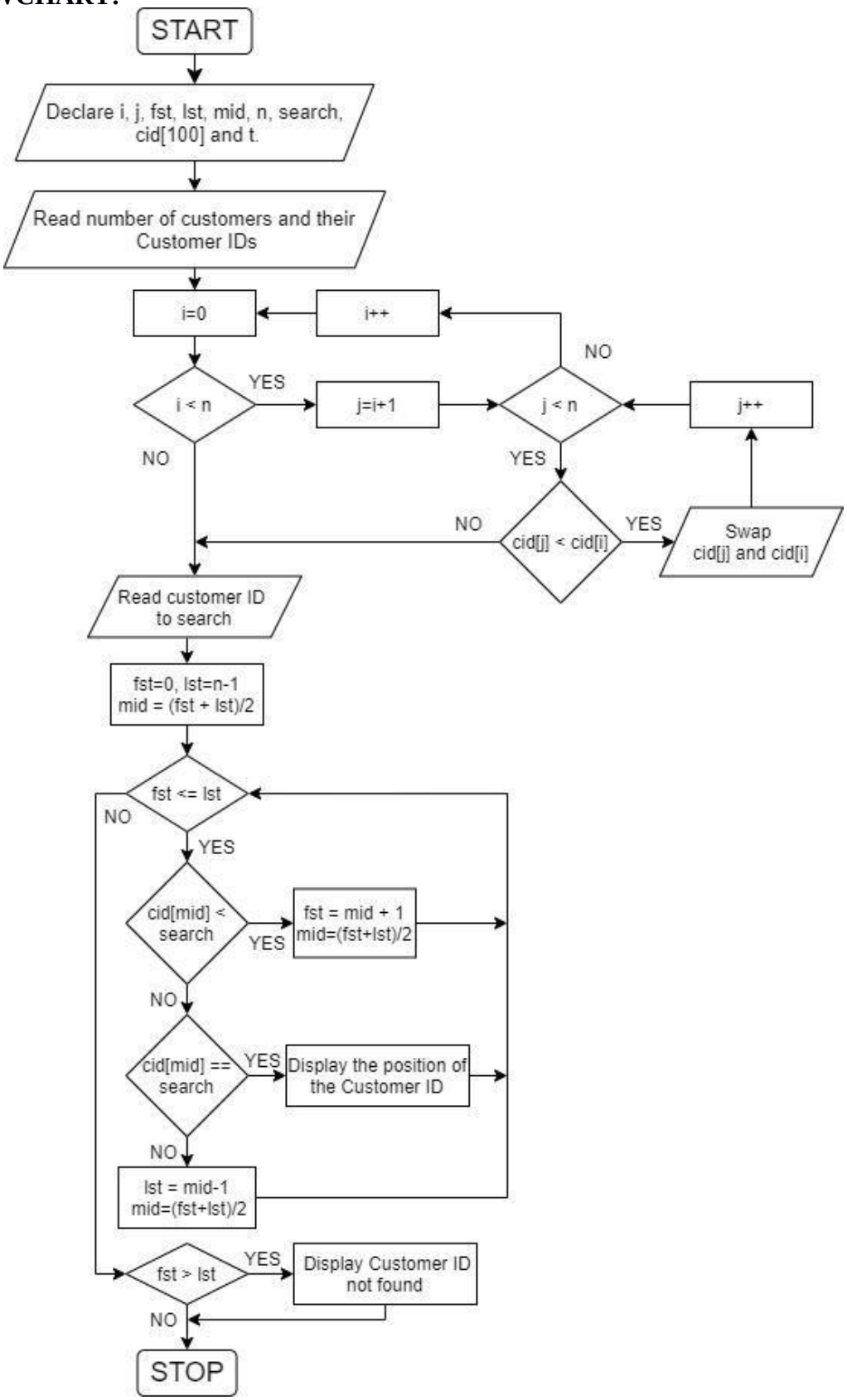
Step 6: If cid[j]<cid[i], then t = cid[i], cid[i] = cid[j] and cid[j] = t. Else, Exit the loop.

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

Step 8: Introduce a while loop that continues if fst<=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.

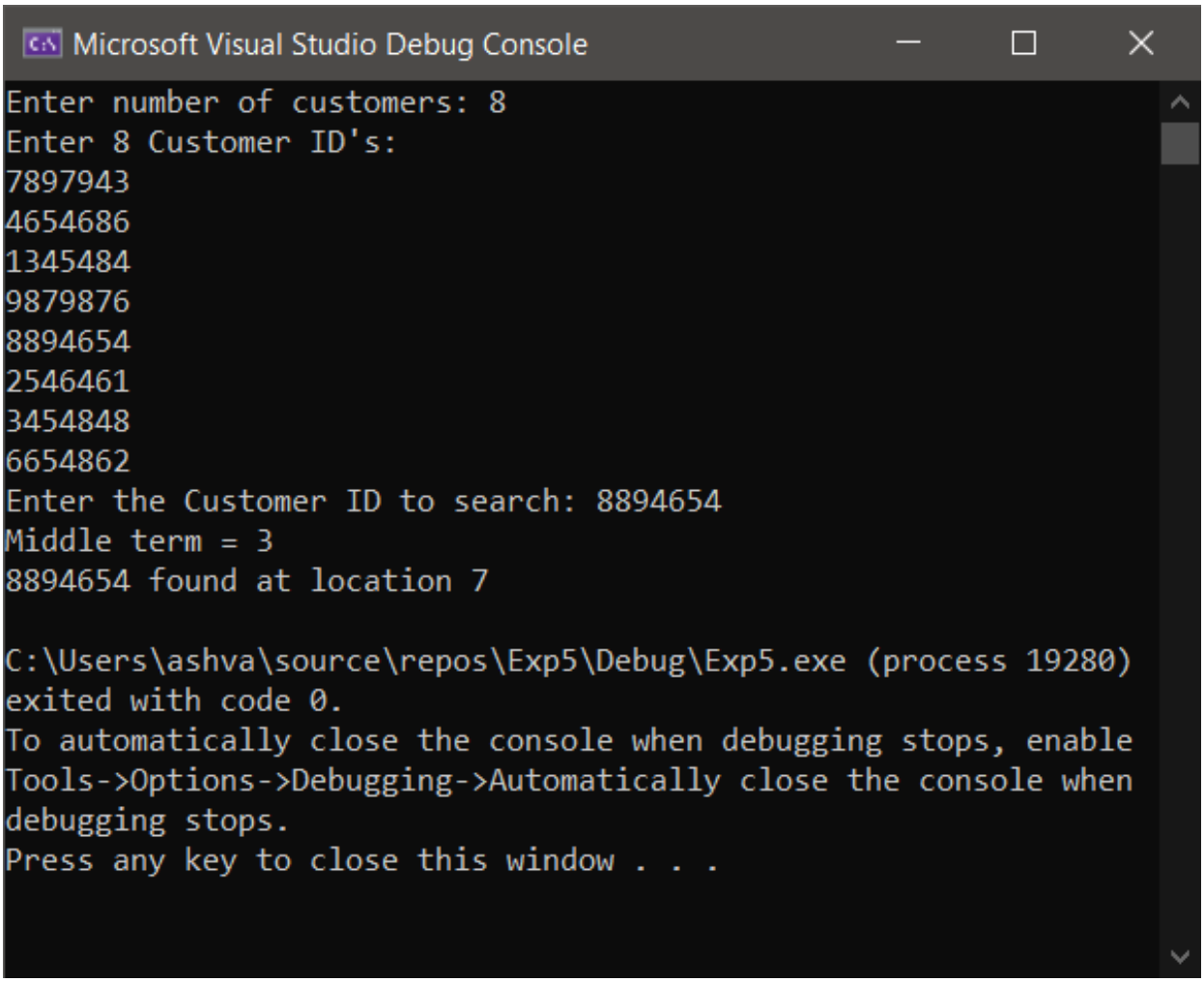
FLOWCHART:


START
Declare i, j, fst, lst, mid, n, search, cid[100] and t.
Read number of customers and their Customer IDs
i=0
i++
i < n
YES
j=i+1
j < n
NO
j++
NO
YES
cid[j] < cid[i]
NO
YES
Swap cid[j] and cid[i]
Read customer ID to search
fst=0, lst=n-1
mid = (fst + lst)/2
fst <= lst
NO
YES
cid[mid] < search
YES
fst = mid + 1
mid=(fst+lst)/2
NO
cid[mid] == search
YES
Display the position of the Customer ID
NO
lst = mid-1
mid=(fst+lst)/2
fst > lst
YES
Display Customer ID not found
NO
STOP

PROGRAM:

```
//Exp 5 – 1D Arrays
#include <stdio.h>
int main()
{
    int i, j, fst, lst, mid, n, search, cid[100], t;
    //Input values
    printf("Enter number of customers: ");
    scanf("%d", &n);
    printf("Enter %d Customer ID's:\n", n);
    for (i = 0; i < n; i++) {
        scanf("%d", &cid[i]);
    }
    //Binary Sort
    for (i = 0; i < n; i++) {
        for (j = i + 1; j < n; j++) {
            if (cid[j] < cid[i]) {
                t = cid[i];
                cid[i] = cid[j];
                cid[j] = t;
            }
        }
    }
    //Binary Search
    printf("Enter the Customer ID to search: ");
    scanf("%d", &search);
    fst = 0;
    lst = n - 1;
    mid = (fst + lst) / 2;
    printf("Middle term = %d\n", mid);
    while (fst <= lst) {
        if (cid[mid] < search) {
            fst = mid + 1;
        }
        else if (cid[mid] == search) {
            printf("%d found at location %d\n", search, mid + 1);
            break;
        }
        else {
            lst = mid - 1;
        }
        mid = (fst + lst) / 2;
    }
    if (fst > lst) {
        printf("%d not found\n", search);
    }
    return 0;
}
```

OUTPUT:


Microsoft Visual Studio Debug Console
Enter number of customers: 8
Enter 8 Customer ID's:
7897943
4654686
1345484
9879876
8894654
2546461
3454848
6654862
Enter the Customer ID to search: 8894654
Middle term = 3
8894654 found at location 7
C:\Users\ashva\source\repos\Exp5\Debug\Exp5.exe (process 19280) exited with code 0.
To automatically close the console when debugging stops, enable Tools->Options->Debugging->Automatically close the console when debugging stops.
Press any key to close this window . . .