	Bansilal Ramnath Agarwal Charitable Trust's Vishwakarma Institute of Information Technology **Department of Artificial Intelligence and Data Science**

Name: Siddhesh Dilip Khairnar		
Class: SY	Division: B	Roll No: 272028
Semester: III	Academic Year: 2022-2023	
Subject Name & Code: Data Structure, ADUA21202		
Title of Assignment: Create a database using array of structures and perform following operations on it: i. Add record ii. Display Database iii. Search record (binary search) iv. Delete record		

Assignment No.- 3

PAGE NO.:
DATE: / /

DS Assignment - 3

Name: Siddhesh Dilip Khairnar
Roll no: 272028

Aim: To perform operation on database created using array of structure

Problem statement: create a database & using array of structure and Perform following operation on it:

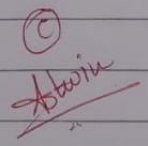
(i) Add record
(ii) Display database
(iii) search record
(iv) Delete record.

Background: Array Database are a class of NO SQL database that store, manage and analyze data whose natural structure are arrange. we have create 4 different function to perform add, display, search and delete operation. At last we have to call each function as per requirement in main function

Array: An array is a collection of item of same data type stored at contiguous memory location. It is the simplest datastructure where each data element can be accessed directly by only using its index number.

Software requirement:
online compiler or Any IDE

Conclusion:
Thus, we successfully created a database using array of structure & Performed add, display, search (binary search) operation



Program:

```c
#include <stdio.h>

void display(int arr[], int size)
{
    for(int i=0; i<size; i++)
    {
        printf("%d ",arr[i]);
    }
    printf("\n");
}

int insert(int arr[], int size, int capacity)
{
    int element, index;
    printf("Enter element: ");
    scanf("%d",&element);
    printf("Enter index: ");
    scanf("%d",&index);
    if(size>capacity)
    {
        return -1;
    }
    for (int i = size-1; i>=index; i--)
    {
        arr[i+1] = arr[i];
    }
    arr[index] = element;
    return 1;
}
```

```c
}

void delete(int arr[],int size)
{
    int index;
    printf("Enter the index of the number you want to delete: ");
    scanf("%d",&index);
    for(int i=index; i<size; i++)
    {
        arr[i] = arr[i+1];
    }
}

int BinarySearch(int arr[],int size)
{
    int element;
    int low = 0, high = size-1, mid;

    printf("Enter the element to be searched: ");
    scanf("%d",&element);
    while(low<=high)
    {
        mid = (low+high)/2;
        if(arr[mid]==element)
        {
            return mid;
        }
        if(arr[mid]<element)
        {
```

```
            low = mid+1;
        }
        else
        {
            high = mid-1;
        }
    }
    return -1;
}

int main()
{
    int a[10] = {3,12,18,41,67,83,90,99};
    int size = 8, element, index, n;
    printf("The original array is:\n");
    display(a,size);

    insert(a,size,10);
    size++;
    printf("The array after insertion is:\n");
    display(a,size);

    delete(a,size);
    size--;
    printf("The array after deletion is:\n");
    display(a,size);

    n = BinarySearch(a,size);
```

```
    printf("The element is at index %d",n);
    return 0;
}
```

Output:

```
The original array is:
3 12 18 41 67 83 90 99
Enter element: 55
Enter index: 4
The array after insertion is:
3 12 18 41 55 67 83 90 99
Enter the index of the number you want to delete: 7
The array after deletion is:
3 12 18 41 55 67 83 99
Enter the element to be searched: 83
The element is at index 6
```