**DATA STRUCTURE**

***PROGRAMS:***

***1.Linear Search***

*#include <stdio.h>*

*int linearSearch(int arr[], int n, int x) {*

*for (int i = 0; i < n; i++) {*

*if (arr[i] == x) {*

*return i;*

*}*

*}*

*return -1;*

*}*

*int* *main() {*

*int arr[] = {2, 5, 8, 12, 16, 23, 38, 56, 72, 91};*

*int n = sizeof(arr) / sizeof(arr[0]);*

*int x = 23;*

*int result = linearSearch(arr, n, x);*

*if (result == -1) {*

*printf("Element not found in the array\n");*

*} else {*

*printf("Element found at index: %d\n", result);*

*}*

*return 0;*

*}*

***OUTPUT:***

*Element found at index: 5*

***2.Binary Search***

*#include <stdio.h>*

*int binarySearch(int arr[], int l, int r, int x) {*

*while (l <= r) {*

*int mid = l + (r - l) / 2;*

*if (arr[mid] == x)*

*return mid;*

*if (arr[mid] < x)*

*l = mid + 1;*

*else*

*r = mid - 1;*

*}*

*return -1;*

*}*

*int* *main() {*

*int arr[] = {2, 3, 4, 10, 40};*

*int n = sizeof(arr) / sizeof(arr[0]);*

*int x = 10;*

*int result = binarySearch(arr, 0, n - 1, x);*

*if (result == -1)*

*printf("Element* *not present in the array");*

*else*

*printf("Element found at index %d", result);*

*return 0;*

*}*

***OUTPUT:***

*Element found at index: 3*

***3.Write a C program to implement*** ***following operations.***

***a. Traverse***

***b. Search***

***c. Insert***

***d. Delete***

***e. Update***

***a. Traverse***

*#include <stdio.h>*

*int* *main() {*

*int arr[] = {1, 2, 3, 4, 5};*

*int size = sizeof(arr) / sizeof(arr[0]);*

*for (int i = 0; i < size; i++) {*

*printf("%d ", arr[i]);*

*}*

*return 0;*

*}*

***OUTPUT:***

*1 2 3 4 5*

***b. Search***

*#include <stdio.h>*

*int* *search(int arr[], int n, int x) {*

*int i;*

*for (i = 0; i < n; i++) {*

*if (arr[i] == x) {*

*return i;*

*}*

*}*

*return -1;*

*}*

*int* *main() {*

*int arr[] = {2, 3, 4, 10, 40};*

*int x = 10;*

*int n = sizeof(arr) / sizeof(arr[0]);*

*int result =* *search(arr, n, x);*

*if (result == -1) {*

*printf("Element is not present in array\n");*

*} else {*

*printf("Element is present at index %d\n", result);*

*}*

*return 0;*

*}*

***OUTPUT:***

*Element is present at index :3*

***C. Insert***

*#include <stdio.h>*

*int* *main() {*

*int* *array[5] = {1, 2, 3, 4, 5};*

*int position = 2;*

*int number\_to\_insert = 10;*

*int array\_size = 5;*

*for (int i = array\_size - 1; i >= position; i--) {*

*array[i] =* *array[i - 1];*

*}*

*array[position] = number\_to\_insert;*

*for (int i = 0; i < array\_size; i++) {*

*printf("%d ", array[i]);*

*}*

*return 0;*

*}*

***OUTPUT:***

*1 2 10 3 4*

***d. Delete***

*#include <stdio.h>*

*void deleteElement(int arr[], int size, int index) {*

*if (index < 0 || index >= size) {*

*printf("Invalid index. Deletion failed.\n");*

*return;*

*}*

*for (int i = index; i < size - 1; i++) {*

*arr[i] = arr[i + 1];*

*}*

*printf("Element at index %d deleted successfully.\n", index);*

*}*

*int* *main() {*

*int arr[] = {1, 2, 3, 4, 5};*

*int size = 5;*

*int index = 2;*

*printf("Original Array: ");*

*for (int i = 0; i < size; i++) {*

*printf("%d ", arr[i]);*

*}*

*deleteElement(arr, size, index);*

*size--;*

*printf("Array after deletion: ");*

*for (int i = 0; i < size; i++) {*

*printf("%d ", arr[i]);*

*}*

*return 0;*

*}*

***OUTPUT:***

*Original Array: 1 2 3 4 5 Element at index 2 deleted successfully.*

*Array after deletion: 1 2 4 5*

***e. Update***

*#include <stdio.h>*

*int* *main() {*

*int arr[] = {10, 20, 30, 40, 50};*

*int index = 2;*

*int new\_number = 35;*

*arr[index] = new\_number;*

*for (int i = 0; i < 5; i++) {*

*printf("%d ", arr[i]);*

*}*

*return 0;*

*}*

***OUTPUT:***

*10 20 35 40 50*

***4.Write a recursive function to calculate the factorial of a number****.*

*#include <stdio.h>*

*int* *factorial(int n) {*

*if (n == 0) {*

*return 1;*

*} else {*

*return n \** *factorial(n - 1);*

*}*

*}*

*int* *main() {*

*int number;*

*printf("Enter a number: ");*

*scanf("%d", &number);*

*int result = factorial(number);*

*printf("Factorial of %d is %d\n", number, result);*

*return 0;*

*}*

***OUTPUT:***

*Enter a number: 5*

*Factorial of 5 is 120*

***5.Write a program to find duplicate element in an array.***

*#include <stdio.h>*

*int* *main() {*

*int arr[] = {1, 2, 3, 4, 2, 5, 6, 3};*

*int size = sizeof(arr) / sizeof(arr[0]);*

*printf("Duplicate elements in the array are: ");*

*for (int i = 0; i < size; i++) {*

*for (int j = i + 1; j < size; j++) {*

*if (arr[i] == arr[j]) {*

*printf("%d ", arr[j]);*

*break;*

*}*

*}*

*}*

*return 0;*

*}*

***OUTPUT:***

*Duplicate elements in the array are: 2 3*

***6.Write a program to find Max and Min from an array*** ***elements.***

*#include <stdio.h>*

*int* *main() {*

*int arr[] = {10, 5, 20, 8, 15};*

*int n = sizeof(arr) / sizeof(arr[0]);*

*int max = arr[0];*

*int min = arr[0];*

*for (int i = 1; i < n; i++) {*

*if (arr[i] > max) {*

*max = arr[i];*

*}*

*if (arr[i] < min) {*

*min = arr[i];*

*}*

*}*

*printf("Maximum element in the array: %d\n", max);*

*printf("Minimum element in the array: %d\n", min);*

*return 0;*

*}*

***OUTPUT:***

*Maximum element in the array: 20*

*Minimum element in the array: 5*

***7.******Given a number n.the task is to print the Fibonacci series and the sum of series using recursion.***

***Input: n=10***

***Output: Fibonacci series***

***0,1,1,2,3,5,8,13,21,34***

***Sum:88***

*#include <stdio.h>*

*int fibonacci(int n) {*

*if (n <= 1)*

*return n;*

*return fibonacci(n - 1) + fibonacci(n - 2);*

*}*

*int* *main() {*

*int n = 10;*

*int sum = 0;*

*printf("Fibonacci Series:\n");*

*for (int i = 0; i < n; i++) {*

*printf("%d, ", fibonacci(i));*

*sum += fibonacci(i);*

*}*

*printf("\nSum: %d\n", sum);*

*return 0;*

*}*

***OUTPUT:***

*Fibonacci Series:*

*0, 1, 1, 2, 3, 5, 8, 13, 21, 34,*

*Sum: 88*

***8.You are given an array arr in increasing order.Find the element x from arr using*** ***binary search.***

***Example*** ***1:arr******={1,5,6,7,9,10******},X=6***

***Output:Element found a location 2***

***Example*** ***2:arr******={1,5,6,7,9,10******},X=11***

***Output:Element not found at location 2***

*#include <stdio.h>*

*int binarySearch(int arr[], int l, int r, int x) {*

*while (l <= r) {*

*int mid = l + (r - l) / 2;*

*if (arr[mid] == x)*

*return mid;*

*if (arr[mid] < x)*

*l = mid + 1;*

*else*

*r = mid - 1;*

*}*

*return -1;*

*}*

*int* *main() {*

*int arr[] = {1, 5, 6, 7, 9, 10};*

*int n = sizeof(arr) / sizeof(arr[0]);*

*int x = 6;*

*int result = binarySearch(arr, 0, n - 1, x);*

*if (result == -1)*

*printf("Element not found\n");*

*else*

*printf("Element found at location %d\n", result);*

*x = 11;*

*result = binarySearch(arr, 0, n - 1, x);*

*if (result == -1)*

*printf("Element not found\n");*

*else*

*printf("Element found at location %d\n", result);*

*return 0;*

*}*

***OUTPUT:***

Element found at location 2

Element not found