**1)Write a c program to implement following operations**

**a)Traverse**

**code**

**#include <stdio.h>**

**int main() {**

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

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

**printf("Array elements:\n");**

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

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

**}**

**printf("\n");**

**return 0;**

**}**

**Output: Array elements:**

**1 2 3 4 5**

**b)Search**

**code**

**#include <stdio.h>**

**int linearSearch(int arr[], int size, int key);**

**int main() {**

**int arr[] = {10, 23, 45, 78, 34, 12, 67};**

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

**int key, index;**

**printf("Enter element to search: ");**

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

**index = linearSearch(arr, size, key);**

**if (index != -1) {**

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

**} else {**

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

**}**

**return 0;**

**}**

**int linearSearch(int arr[], int size, int key) {**

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

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

**return i;**

**}**

**}**

**return -1;**

**}**

**Output: Enter element to search: 78**

**Element found at index 3.**

**c)Insert**

**code**

**#include <stdio.h>**

**void insertElement(int arr[], int \*size, int position, int element);**

**int main() {**

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

**int size = 5;**

**int position, element;**

**printf("Initial Array:\n");**

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

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

**}**

**printf("\n");**

**printf("Enter the position to insert (0-%d): ", size);**

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

**if (position < 0 || position > size) {**

**printf("Invalid position!\n");**

**return 1;**

**}**

**printf("Enter the element to insert: ");**

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

**insertElement(arr, &size, position, element);**

**printf("Modified Array:\n");**

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

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

**}**

**printf("\n");**

**return 0;**

**}**

**void insertElement(int arr[], int \*size, int position, int element) {**

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

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

**}**

**arr[position] = element;**

**(\*size)++;**

**}**

**Output: Initial Array:**

**10 20 30 40 50**

**Enter the position to insert (0-5): 2**

**Enter the element to insert: 35**

**Modified Array:**

**10 20 35 30 40 50**

**d)Delete**

**Code**

**#include <stdio.h>**

**#define MAX\_SIZE 100 // Maximum size of the array**

**void deleteElement(int arr[], int \*size, int position);**

**int main() {**

**int arr[MAX\_SIZE];**

**int size, position;**

**printf("Enter the size of the array (max %d): ", MAX\_SIZE);**

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

**printf("Enter %d elements:\n", size);**

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

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

**}**

**printf("Enter the position of the element to delete (0-%d): ", size - 1);**

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

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

**printf("Invalid position!\n");**

**return 1;**

**}**

**deleteElement(arr, &size, position);**

**printf("Modified Array:\n");**

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

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

**}**

**printf("\n");**

**return 0;**

**}**

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

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

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

**}**

**(\*size)--;**

**}**

**Output: Enter the size of the array (max 100): 6**

**Enter 6 elements:**

**10**

**20**

**30**

**40**

**50**

**60**

**Enter the position of the element to delete (0-5): 3**

**Modified Array:**

**10 20 30 50 60**

**e)Update**

**Code**

**#include <stdio.h>**

**#define MAX\_SIZE 100 // Maximum size of the array**

**void updateElement(int arr[], int size, int position, int newValue);**

**int main() {**

**int arr[MAX\_SIZE]; // Array declaration**

**int size, position, newValue;**

**printf("Enter the size of the array (max %d): ", MAX\_SIZE);**

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

**printf("Enter %d elements:\n", size);**

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

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

**}**

**printf("Enter the position of the element to update (0-%d): ", size - 1);**

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

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

**printf("Invalid position!\n");**

**return 1;**

**}**

**printf("Enter the new value for the element: ");**

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

**updateElement(arr, size, position, newValue);**

**printf("Modified Array:\n");**

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

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

**}**

**printf("\n");**

**return 0;**

**}**

**void updateElement(int arr[], int size, int position, int newValue) {**

**arr[position] = newValue;**

**}**

**Output: Enter the size of the array (max 100): 5**

**Enter 5 elements:**

**10**

**20**

**30**

**40**

**50**

**Enter the position of the element to update (0-4): 3**

**Enter the new value for the element: 60**

**Modified Array:**

**10 20 30 60 50**

**2) Writing a recursive function to calculate the factorial of a number.**

**Code**

**#include <stdio.h>**

**unsigned long long factorial(int n);**

**int main() {**

**int number;**

**unsigned long long fact;**

**printf("Enter a positive integer: ");**

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

**if (number < 0) {**

**printf("Error: Factorial of negative number is undefined.\n");**

**} else {**

**fact = factorial(number);**

**printf("Factorial of %d = %llu\n", number, fact);**

**}**

**return 0;**

**}**

**unsigned long long factorial(int n) {**

**factorial of 0 is 1**

**if (n == 0) {**

**return 1;**

**}**

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

**else {**

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

**}**

**}**

**Output: Enter a positive integer: 5**

**Factorial of 5 = 120**

**3) Write a c program to find duplicate element in an array.**

**Code**

**#include <stdio.h>**

**#define MAX\_SIZE 100**

**int main() {**

**int arr[MAX\_SIZE];**

**int size;**

**printf("Enter the size of the array (max %d): ", MAX\_SIZE);**

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

**printf("Enter %d elements:\n", size);**

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

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

**}**

**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[i]);**

**break;**

**}**

**}**

**}**

**printf("\n");**

**return 0;**

**}**

**Output: Enter the size of the array (max 100): 8**

**Enter 8 elements**

**3 4 5 6 3 4 7 8**

**Duplicate elements in the array are: 3 4**

**4)Write a C program to find max and min from an array elements.**

**Code**

**#include <stdio.h>**

**#define MAX\_SIZE 100 // Maximum size of the array**

**int main() {**

**int arr[MAX\_SIZE];**

**int size;**

**printf("Enter the size of the array (max %d): ", MAX\_SIZE);**

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

**printf("Enter %d elements:\n", size);**

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

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

**}**

**int max = arr[0];**

**int min = arr[0];**

**for (int i = 1; i < size; 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: Enter the size of the array (max 100): 5**

**Enter 5 elements:**

**12 34 10 56 92**

**Maximum element in the array: 92**

**Minimum element in the array: 10**

**5) Given a number n. The task is to print the fibonacci series and the sum of the series using recursion**

**Input: n=10**

**Output: fibonacci series**

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

**Sum:88**

**Code**

**#include <stdio.h>**

**int fibonacci(int n);**

**int fibonacciSum(int n);**

**int main() {**

**int n;**

**printf("Enter the number of terms in the Fibonacci series: ");**

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

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

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

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

**}**

**printf("\n");**

**printf("Sum of Fibonacci Series: %d\n", fibonacciSum(n))**

**return 0;**

**}**

**int fibonacci(int n) {**

**if (n <= 1) {**

**return n;**

**} else {**

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

**}**

**}**

**int fibonacciSum(int n) {**

**if (n <= 0) {**

**return 0;**

**} else {**

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

**}**

**}**

**Output: Enter the number of terms in the Fibonacci series: 10**

**Fibonacci Series:**

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

**Sum of Fibonacci Series: 88**

**6) You are given an array in increasing order find the element**

**X from arr using binary**

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

**Output:element found at location 2**

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

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

**Code**

**#include <stdio.h>**

**int binarySearch(int arr[], int low, int high, int x);**

**int main() {**

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

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

**int x;**

**printf("Enter the element to search: ");**

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

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

**if (index != -1) {**

**printf("Element found at location %d\n", index + 1);**

**} else {**

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

**}**

**return 0;**

**}**

**int binarySearch(int arr[], int low, int high, int x) {**

**while (low <= high) {**

**int mid = low + (high - low) / 2;**

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

**return mid;**

**}**

**if (arr[mid] < x) {**

**low = mid + 1;**

**} else { // If x is smaller, ignore right half**

**high = mid - 1;**

**}**

**}**

**return -1;**

**}**

**Output: Enter the element to search: 6**

**Element found at location 3**