1.

#include<iostream>

using namespace std;

int main()

{

int arr[6];

cout << "Input array elements: ";

for (int i = 0; i<6; i++) {

cin>> arr[i];

}

int max = arr[0], min = arr[0];

for (int i = 1; i < 6; i++) {

if (arr[i] > max) {

max = arr[i];

}

if (arr[i] < min) {

min = arr[i];

}

}

cout << "Maximum: "<< max << endl;

cout << "Minimum: "<< min << endl;

}

2.

#include <iostream>

using namespace std;

void findHighestTemperature(int temps[], int size) {

int maxTemp = temps[0];

int count = 1;

// Find the maximum temperature and count how many times it appears

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

if (temps[i] > maxTemp) {

maxTemp = temps[i];

count = 1;

} else if (temps[i] == maxTemp) {

count++;

}

}

cout << "Highest Temperature: " << maxTemp << "°C" << endl;

cout << "It occurred " << count << " times." << endl;

}

int main() {

const int DAYS = 7;

int temperatures[DAYS];

cout << "Enter the temperatures for the last 7 days:" << endl;

for (int i = 0; i < DAYS; i++) {

cout << "Day " << (i + 1) << ": ";

cin >> temperatures[i];

}

findHighestTemperature(temperatures, DAYS);

return 0;

}

3.

#include <iostream>

using namespace std;

int main() {

int n=5;

int arr[n], size, pos;

cout<<"Input array elements:"<<endl;

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

cin>> arr[i];

}

cout<<"Input position to delete:"<<endl;

cin>> pos;

// Adjust for zero-based index

pos--;

// Shift elements to the left from the specified position

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

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

}

n--;

cout<<"Array after deletion: ";

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

cout<< arr[i]<<" ";

}

cout<< endl;

return 0;

}

5.

#include <iostream>

using namespace std;

int main() {

int original\_array[] = {2, 5, 8, 1, 10, 3, 7};

int size = sizeof(original\_array) / sizeof(original\_array[0]);

int threshold = 5;

// Create an array to store the filtered elements

int filtered\_array[size];

int filtered\_size = 0;

// Iterate over the original array

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

if(original\_array[i] > threshold) {

// Add element to filtered array if it is greater than the threshold

filtered\_array[filtered\_size] = original\_array[i];

filtered\_size++;

}

}

cout << "Filtered array: ";

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

cout << filtered\_array[i] << " ";

}

return 0;

}

9.

#include <iostream>

using namespace std;

int main() {

int array[] = {1, 2, 5, 7, 9, 10};

int size = sizeof(array) / sizeof(array[0]);

int subarray[] = {5, 7}, a=0;

for(int i = 0; i < size - 1; i++)

{

if(array[i] == subarray[0] && array[i + 1] == subarray[1]) {

a++;

}

}

if(a>0) {

cout << "Found" << endl;

}

else {

cout << "Not found" << endl;

}

return 0;

}