**Exercises**

Q1-Write a C++ program to find the largest three elements in an array.

#include<iostream>

using namespace std;

void three\_largest(int arr[], int arr\_size)

{

int i, first, second, third;

if (arr\_size < 3)

{

cout << "Invalid Input";

}

third = first = second = INT\_MIN;

for (i = 0; i < arr\_size ; i ++)

{

if (arr[i] > first)

{

third = second;

second = first;

first = arr[i];

}

else if (arr[i] > second)

{

third = second;

second = arr[i];

}

else if (arr[i] > third)

third = arr[i];

}

cout << "\nThree largest elements are: " <<first <<", "<< second <<", "<< third;

}

int main()

{

int nums[] = {7, 12, 9, 15, 19, 32, 56, 70};

int n = sizeof(nums)/sizeof(nums[0]);

cout << "Original array: ";

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

cout << nums[i] <<" ";

three\_largest(nums, n);

return 0;

}

Q2- Write a C++ program to find the most occurring element in an array of integers.

#include<iostream>

using namespace std;

void most\_occurred\_number(int nums[], int size)

{

int max\_count = 0;

cout << "\nMost occurred number: ";

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

{

int count=1;

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

if (nums[i]==nums[j])

count++;

if (count>max\_count)

max\_count = count;

}

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

{

int count=1;

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

if (nums[i]==nums[j])

count++;

if (count==max\_count)

cout << nums[i] << endl;

}

}

int main()

{

int nums[] = {4, 5, 9, 12, 9, 22, 45, 7};

int n = sizeof(nums)/sizeof(nums[0]);

cout << "Original array: ";

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

cout << nums[i] <<" ";

most\_occurred\_number(nums, n);

}

Q3-Write a C++ program to find and print all unique elements of a given array of integers.

#include <iostream>

using namespace std;

int main()

{

int array1[] = {1, 5, 7, 5, 8, 9, 11, 11, 2, 5, 6};

int s1 = sizeof(array1)/sizeof(array1[0]);

cout << "Original array: ";

for (int i=0; i < s1; i++)

cout << array1[i] <<" ";

cout <<"\nUnique elements of the said array: ";

for (int i=0; i<s1; i++)

{

int j;

for (j=0; j<i; j++)

if (array1[i] == array1[j])

break;

if (i == j)

cout << array1[i] << " ";

}

return 0;

}

Q4-Write a C++ program to separate 0s and 1s from a given array of values 0 and 1.

#include<iostream>

using namespace std;

void segregateEvenOdd(int nums[], int n)

{

int ctr = 0;

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

if (nums[i] == 0)

ctr++;

}

for (int i = 0; i < ctr; i++)

nums[i] = 0;

for (int i = ctr; i < n; i++)

nums[i] = 1;

}

int main()

{

int nums[] = {0, 1, 0, 0 , 1, 1, 1, 0, 1, 0};

int n = sizeof(nums)/sizeof(nums[0]);

cout << "Original array: ";

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

cout << nums[i] <<" ";

segregateEvenOdd(nums, n);

printf("\nArray after divided: ");

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

cout << nums[i] <<" ";

return 0;

}

Q5-Write a C++ program to find the two repeating elements in a given array of integers.

#include <iostream>

using namespace std;

int main()

{

int nums[] = {1, 2, 3, 5, 5, 7, 8, 8, 9, 9, 2};

int i, j;

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

cout << "Original array: ";

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

cout << nums[i] <<" ";

cout << "\nRepeating elements: ";

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

for(j = i+1; j < size; j++)

if(nums[i] == nums[j])

cout << nums[i] << " ";

return 0;

}