Q1. Using STL Array gets and sets a reference to an element based on a given index.

#include <iostream>

#include <array>

using namespace std;

int& gets(array <int, 5> &arr, int index);

int main()

{

array <int, 5> arr = {10, 20, 30, 40, 50};

int index = 0, value = 0;

cout<<"Enter index = ";

cin>>index;

cout<<"Enter value = ";

cin>>value;

gets(arr, index) = value;

for(int x : arr) cout<<x<<" ";

}

int& gets(array <int, 5> &arr, int index)

{

return arr[index];

}

Q2. Using STL Array returns the total number of elements in the array.

#include <iostream>

#include <array>

using namespace std;

int main()

{

array <int, 5> arr = {10, 20, 30, 40, 50};

cout<<"Total elements = "<<arr.size();

}

Q3. Find the first and last element using the STL array.

#include <iostream>

#include <array>

using namespace std;

int main()

{

array <int, 5> arr = {10, 20, 30, 40, 50};

cout<<"First element = "<<arr.front()<<endl;

cout<<"Last element = "<<arr.back()<<endl;

}

Q4. Returns the element from the given index using the STL array.

#include <iostream>

#include <array>

using namespace std;

int main()

{

array <int, 5> arr = {10, 20, 30, 40, 50};

cout<<"0 -> "<<arr.at(0)<<endl;

cout<<"1 -> "<<arr.at(1)<<endl;

cout<<"2 -> "<<arr.at(2)<<endl;

cout<<"3 -> "<<arr.at(3)<<endl;

cout<<"4 -> "<<arr.at(4)<<endl;

}

Q5. C++ STL program to demonstrate example of array::rbegin() and array::rend() functions

#include <iostream>

#include <array>

#include <iterator>

using namespace std;

int main()

{

array <int, 5> arr = {10, 20, 30, 40, 50};

array <int, 5>::reverse\_iterator rit = arr.rbegin();

cout<<"using rbegin() -> ";

while( rit != arr.rend())

{

cout<<\*rit<<" ";

rit++;

}

rit = arr.rend();

rit = rit - 1;

cout<<endl<<"using rend() -> ";

while(true)

{

cout<<\*rit<<" ";

if(rit == arr.rbegin()) break;

--rit;

}

}

Q6. Using STL to check whether an array is empty or not.

#include <iostream>

#include <array>

using namespace std;

int main()

{

array <int, 5> arr;

if(!arr.empty())

cout<<"array is empty";

else

cout<<"array is not empty";

}

Q7. Sort an array in ascending order using sort() function in C++ STL

#include <iostream>

#include <array>

#include <algorithm>

using namespace std;

int main()

{

array <int, 5> arr = {50, 10, 40, 30, 20};

sort(arr.begin(), arr.end());

for(int x : arr) cout<<x<<" ";

}

Q8. Sort an array in descending order using sort() function in C++ STL

#include <iostream>

#include <array>

#include <algorithm>

using namespace std;

int main()

{

array <int, 5> arr = {50, 10, 40, 30, 20};

sort(arr.begin(), arr.end(), greater<int>());

for(int x : arr) cout<<x<<" ";

}

Q9. C++ program to find the integers which come an odd number of times in an array using C++ STL.

#include <iostream>

#include <array>

using namespace std;

int main()

{

array <int, 14> arr = {1, 2, 3, 2, 3, 4, 3, 4, 5, 4, 5, 4, 6, 4};

int arr\_size = arr.size();

for(int curr = 0; curr < arr\_size; curr++)

{

int prv = 0;

bool flag = true;

if(flag)

{

while(prv < curr)

{

if(arr[prv] == arr[curr])

{

flag = false;

break;

}

else

{

prv = prv + 1;

}

}

}

if(flag)

{

int cont = 1;

int onetime = 1;

for(int next = curr + 1; next < arr\_size; next++)

{

if(arr[curr] == arr[next])

{

cont = cont + 1;

onetime = 0;

}

}

if(onetime)

{

cout<<arr[curr]<<" -> "<<cont<<endl;

}

else

{

if( (cont%2) > 0 )

{

cout<<arr[curr]<<" -> "<<cont<<endl;

}

}

}

}

}

Q10. Given an integer array nums , return an array answer such that answer[i] is equal to

the product of all the elements of nums except nums[i] .

#include <iostream>

#include <array>

using namespace std;

int sum (array <int, 5>& nums, int i);

int main()

{

array <int, 5> nums = {50, 10, 40, 30, 20};

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

{

int answer = sum(nums, i);

cout<<"nums["<<i<<"] -> "<<answer<<endl;

}

}

int sum (array <int, 5>& nums, int i)

{

int answer = 0;

for(int j = 0; j < nums.size(); j++)

{

if(j == i)

{

continue;

}

else

{

answer = answer + nums[j];

}

}

return answer;

}