1olculu massivler 3-cu variant

1)

#include <iostream>

#include <cmath>

using namespace std;

int main()

{

int X[15]={2,4,6,8,10,12,14,16,18,20,22,26,26,28,30};

int Z[15]={2,2,3,8,5,6,14,8,9,20,20,26,13,14,15};

int Y[15];

int index=0;

for(int i=0;i<15;i++){

if(X[i]==Z[i] && X[i]!=0 ){

Y[index++]=X[i];

}

}

for(int i=0;i<15;i++){

if(Y[i]!=0){

cout<<Y[i]<<" ";

}

}

return 0;

}

2) 8 elementdən ibarət A və B massivləri verilir. Elementlləri qaydası ilə hesablanan C massivinin ən böyük ədədini tapmalı

#include <iostream>

using namespace std;

int main()

{

float A[8]={2,4,6,8,10,12,14,1};

float B[8]={2,2,3,8,5,6,14,8};

float C[8];

for(int i=0;i<8;i++){

C[i]=(A[i]-B[i])/(A[i]\*B[i]);

}

for(int i=0;i<8;i++){

cout<<C[i]<<" ";

}

float maxElement=C[0];

for(int i=0;i<8;i++){

if(C[i]>maxElement){

maxElement=C[i];

}

}

cout<<endl;

cout<<"Max element : "<<maxElement;

return 0;

}

3) A massivindən elementlərinin işarəsi onunla əks olan B mas­si­vi­ni düzəltməli və B mas­si­vinin elementləri cəmini hesablamalı

#include <iostream>

#include <cmath>

using namespace std;

int main()

{

float A[5]={2,-7,6,8,-11};

float B[5];

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

B[i]=A[i]\*(-1);

}

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

cout<<B[i]<<" ";

}

int sum=0;

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

sum+=B[i];

}

cout<<endl;

cout<<"Sum : "<<sum;

return 0;

}

4) A(n) massivində maksimum elementdən sonra yerləşən elementlərin cəmini tapın.

#include <iostream>

using namespace std;

int main()

{

float A[8]={2,4,6,5,100,12,14,1};

int maxIndex=0;

for(int i=0;i<8;i++){

if(A[i]>A[maxIndex]){

maxIndex=i;

}

}

cout<<endl;

cout<<"Max element : "<<A[maxIndex];

int sum=0;

for(int i=maxIndex+1;i<8;i++){

sum+=A[i];

}

cout<<endl;

cout<<"Sum : "<<sum;

return 0;

}

5) A(10) massivində minimum elementdən əvvəlki elementlərin cəmini tapın.

#include <iostream>

using namespace std;

int main()

{

float A[8]={2,4,6,5,0,12,14,16};

int minIndex=0;

for(int i=0;i<8;i++){

if(A[i]<A[minIndex]){

minIndex=i;

}

}

cout<<endl;

cout<<"min element : "<<A[minIndex];

int sum=0;

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

sum+=A[i];

}

cout<<endl;

cout<<"Sum : "<<sum;

return 0;

}

6) A(10) massivində şərtini ödəyən elementlərin cəmini tapmalı

#include <iostream>

using namespace std;

int main()

{

float A[10]={2,4,6,5,0,12,14,16,18,20};

int b=5;

int c=7;

int sum=0;

for(int i=0;i<10;i++){

if(b<abs(A[i]) && abs(A[i])<c){

sum+=A[i];

};

}

cout<<endl;

cout<<"Sum : "<<sum;

return 0;

}

7) A(n) və B(m) massivləri verilmişdir. Hər ikisində maksimum elementləri tapıb yerlərini dəyişməli

#include <iostream>

using namespace std;

int main()

{

int A[8]={2,4,6,5,100,12,14,1};

int B[5]={1,2,3,19,6};

int maxIndexA=0;

for(int i=0;i<8;i++){

if(A[i]>A[maxIndexA]){

maxIndexA=i;

}

}

int maxIndexB=0;

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

if(B[i]>B[maxIndexB]){

maxIndexB=i;

}

}

int maxElementA=A[maxIndexA];

A[maxIndexA]=B[maxIndexB];

B[maxIndexB]=maxElementA;

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

cout<<B[i]<<" ";

}

cout<<endl;

for(int i=0;i<8;i++){

cout<<A[i]<<" ";

}

return 0;

}

8) A(n) massivində ən kiçik və ən böyük elementlər arasında yerləşən elementlərin hasilini tapın.

#include <iostream>

using namespace std;

int main()

{

int A[8]={2,0,6,5,100,12,14,1};

int B[5]={1,2,3,19,6};

int maxIndexA=0;

int minIndexA=0;

for(int i=0;i<8;i++){

if(A[i]>A[maxIndexA]){

maxIndexA=i;

}

if(A[i]<A[minIndexA]){

minIndexA=i;

}

}

int product=1;

cout<<endl;

if(minIndexA<maxIndexA){

for(int i=minIndexA+1;i<maxIndexA;i++){

product\*=A[i];

}

}

else{

for(int i=maxIndexA+1;i<minIndexA;i++){

product\*=A[i];

}

}

cout<<product;

return 0;

}

9) X(n) massivinin elementlərini azalma sırasına görə nizamlamalı

//yalniz musbet ededler uchun ishleyir

int main ()

{

int size = 8;

int X[size] = { 0, 18, 6, 0, 100, 12, 14, 1 };

int B[size] = { };

int maxIndexX = 0;

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

{

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

{

if (X[i] > X[maxIndexX] && X[i] != false)

{

maxIndexX = i;

}

}

B[i] = X[maxIndexX];

X[maxIndexX] = false;

}

for (int i = 0; i < 8; i++)

{

cout << B[i] << " ";

}

return 0;}

10) X(20) massivinin birinci yarısının ədədi ortasını, ikinci yarısı­nın həndəsi ortasını tapmalı

#include <iostream>

#include<cmath>

using namespace std;

int main ()

{

int size = 20;

int X[size] = { 5, 18, 6, 8, 100, 12, 14, 1,12,3,2,1,3,1,90,2,1,19,89,43 };

int avg=0;

for (int i = 0; i < size/2; i++)

{

avg+=X[i];

}

avg/=10;

int power=1;

for (int i = size/2; i < size; i++)

{

power\*=X[i];

}

power=pow(power,1/10);

cout<<"pow->"<<power<<endl;

cout<<"avg->"<<avg<<endl;

return 0;

}

11) X(n) massivində ilk və son sıfıra bərabər elementləri arasındakı məsafəni (elementlərin sayını) tapmalı

#include <iostream>

#include<cmath>

using namespace std;

int main ()

{

int size = 20;

int X[size] = { 5, 18, 0, 8, 100, 12, 14, 1,12,3,2,1,3,1,90,0,1,19,89,43 };

int firstZeroIndex;

int lastZeroIndex;

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

{

if(X[i]==0){

firstZeroIndex=i;

break;

}

}

for (int i = size-1; i >-1; i--)

{

if(X[i]==0){

lastZeroIndex=i;

break;

}

}

cout<<lastZeroIndex-firstZeroIndex-1<<endl;

return 0;

}

12) X(n) massivində yalnız mənfi elementlərin içərisindən minimum elementi tapmalı

#include <iostream>

using namespace std;

int main()

{

int X[8]={2,4,6,-5,0,12,-14,16};

int B[8]={};

int minIndex=0;

for(int i=0;i<8;i++){

if(X[i]<0){

B[i]=X[i];

}

}

for(int i=0;i<8;i++){

if(B[i]<B[minIndex]){

minIndex=i;

}

}

cout<<endl;

cout<<"min element : "<<B[minIndex];

return 0;

}