№1  
#include <iostream>

using namespace std;

int square(int num)

{

num = num \* num;

return num;

}

void couting(int num)

{

std::cout << square(num) << std::endl;

}

int main()

{

int num;

std::cout << "Введите число" << std::endl;

std::cin >> num;

couting(num);

return 0;

}

№2

#include <iostream>

#include <cmath>

#include <stdlib.h>

using namespace std;

float sqrti (float num)

{

if (num < 0)

{

std::cout << "Negative argument" << std::endl;

exit(0);

}

else

{

num = sqrt(num);

}

return num;

}

int main()

{

float num;

std::cout << "Введите число" << std::endl;

std::cin >> num;

std::cout << sqrti(num) << std::endl;

return 0;

}

№3

#include <iostream>

using namespace std;

float multiplyOrAdd (float num1, float num2, bool add = 1)

{

float num;

if (add)

{

num = num1 \* num2;

}

else

{

num = num1 + num2;

}

return num;

}

int main()

{

float num1, num2;

bool add;

std::cout << "Введите первое число" << std::endl;

std::cin >> num1;

std::cout << "Введите второе число" << std::endl;

std::cin >> num2;

std::cout << "Введите 1, если умножение. Введите 0, если сложение" << std::endl;

std::cin >> add;

if (add == 1)

{

std::cout << multiplyOrAdd(num1, num2) << std::endl;

}

else

{

std::cout << multiplyOrAdd(num1, num2, add) << std::endl;

}

return 0;

}

№4

#include <iostream>

using namespace std;

int countOdd(int\* arr, int size)

{

int counter = 0;

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

{

if (arr[i] % 2 == 0)

{

counter++;

}

}

return counter;

}

int main()

{

int n;

int\* arr = new int[n];

std::cout << "Введите длину массива: ";

std::cin >> n;

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

{

std::cout << "Введите число: ";

std::cin >> arr[i];

}

std::cout << countOdd(arr, n) << std::endl;

return 0;

}

№5

#include<iostream>

#include<stdlib.h>

using namespace std;

int sum(int n, int size, int arr[]) {

if (n < 0 || n >= size) {

cout << "Ошибка: индекс n внедопустимогодиапазона." << endl;

exit(0);

}

return arr[n] + n;

}

int main() {

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

int n;

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

cout << "Введите номер элемента в массиве n: ";

cin >> n;

cout << "Сумма n-го элемента массива и n: " << sum(n, size, arr) << endl;

return 0;

}

№6

#include<iostream>

using namespace std;

void change (int n, int arr[]) {

for (int i = 0; i < 4; i++){

if (i == n) {

arr[i] =0;

}

}

n++;

for (int I = 0; I < 4; i++){

cout << arr[i];

}

cout << endl << n;

}

int main() {

int n = 2;

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

change (n, arr);

return 0;

}

№7

#include <iostream>

int compareStrings (const char \*str1, const char \*str2) {

const char \*ptr1 = str1;

while (\*ptr1 != '\0') {

ptr1++;

}

int length1 = ptr1 - str1;

const char \*ptr2 = str2;

while (\*ptr2 != '\0') {

ptr2++;

}

int length2 = ptr2 - str2;

std::cout << length1 << std::endl;

std::cout << length2 << std::endl;

if (length1 == length2) {

return 0;

} else if (length1 < length2) {

return -1;

} else {

return 1;

}

}

Int main() {

const char \*str1 = "Hello";

const char \*str2 = "World";

int result = compareStrings (str1, str2);

std::cout << result << std::endl;

return 0;

}

№8

#include<iostream>

int \*findCommonElements (int \*arr1, int size1, int \*arr2, int size2, int& commonSize) {

static int commonElements[99];

commonSize = 0;

for (int \*ptr1 = arr1; ptr1 < arr1 + size1; ptr1++) {

for (int\*ptr2=arr2; ptr2<arr2+size2; ++ptr2) {

if (\*ptr1==\*ptr2) {

commonElements[commonSize++] =\*ptr1;

break;

}

}

}

returncommonElements;

}

intmain() {

intarr1[] = {1, 2, 3, 4, 5};

intarr2[] = {3, 4, 5, 6, 7};

intcommonSize=0;

int\*commonElements=findCommonElements(arr1, 5, arr2, 5, commonSize);

std::cout<<"Общиеэлементы: ";

for (inti=0; i<commonSize; ++i) {

std::cout<<commonElements[i] <<" ";

}

std::cout<<"\nКоличество: "<<commonSize<<std::endl;

return0;

}

//9

#include<iostream>

voidrotateArray(int\*arr, intsize, intpositions) {

positions=positions%size;

int\*temp=newint[positions];

for (inti=0; i<positions; ++i) {

temp[i] =arr[i];

}

for (inti=0; i<size-positions; ++i) {

arr[i] =arr[i+positions];

}

for (inti=0; i<positions; ++i) {

arr[size-positions+i] =temp[i];

}

delete[]temp;

}

intmain() {

intarr[] = {1, 2, 3, 4, 5, 6, 7};

intsize=sizeof(arr) /sizeof(arr[0]);

intpositions=3;

rotateArray(arr, size, positions);

std::cout<<"Сдвинутыймассив: ";

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

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

}

std::cout<<std::endl;

return0;

}

//10

#include<iostream>

intmergeAndCompare(int\*arr1, intsize1, int\*arr2, intsize2, int\*&mergedArray, int&mergedSize) {

if (size1!=size2) {

return-1;

}

mergedSize=size1+size2;

mergedArray=newint[mergedSize];

for (inti=0; i<size1; ++i) {

mergedArray[i] =arr1[i];

}

for (inti=0; i<size2; ++i) {

mergedArray[size1+i] =arr2[i];

}

return0;

}

intmain() {

intsize1=5;

intsize2=5;

int\*arr1=newint[size1]{1, 2, 3, 4, 5};

int\*arr2=newint[size2]{6, 7, 8, 9, 10};

int\*mergedArray=nullptr;

intmergedSize=0;

intresult=mergeAndCompare(arr1, size1, arr2, size2, mergedArray, mergedSize);

if (result==-1) {

std::cout<<result<<std::endl;

} else {

std::cout<<"Объединённый массив: ";

for (inti=0; i<mergedSize; ++i) {

std::cout<<mergedArray[i] <<" ";

}

std::cout<<"\nРазмер: "<<mergedSize<<std::endl;

}

delete[]arr1;

delete[]arr2;

delete[]mergedArray;

return0;

}