# Часть первая. Работа с одномерными массивами (векторами)

Задание

Написать программу, которая решает задачи, определенные вариантом. Программа должна содержать следующие функции:

 ввод массива пользователем, заполнение массива случайными числами; вывод массива, выполнение задания 1, выполнение задания 2.

# Часть вторая. Работа с двумерными массивами (матрицами)

Задание

Написать программу, которая решает задачу, определенную вариантом. Программа должна содержать следующие функции: ввод матрицы пользователем, заполнение матрицы случайными числами, вывод матрицы на экран, выполнение задания.

# Содержание отчета

Титульный лист, вариант задания, прототипы функций, примеры выполнения заданий с произвольными числами, тексты программы.

Пример программы с одномерным массивом

Задание: В одномерном массиве целых чисел вычислить сумму элементов и найти индекс первого нечетного элемента.

#include "stdafx.h"

const int n = 5;

void in\_massiv(int mas[n]);

void print\_massiv(int mas[n]);

int summa(int mas[n]);

int first\_nechet(int mas[n]);

int first\_nechet2(int mas[n]);// второй вариант поиска индекса нечетного

void main()

{ int mas [n]; int s, numfirst=-1, comanda =6;

 do {

 printf ("\n1 - vvod, 2 - print, 3 - summa, 4 - pervyi nechet, 5 - versia 2, 0 - out\n");

 scanf ("%d",&comanda);

 switch (comanda ){

 case 1: in\_massiv(mas); break;

 case 2: print\_massiv(mas); break;

 case 3: s=summa(mas); printf ("\nSumma massiva = %d", s);break;

 case 4: numfirst = first\_nechet(mas);

 printf ("\nIndex pervogo nechetnogo = %d",numfirst);break;

 case 5: numfirst = first\_nechet2(mas);

 printf ("\nIndex pervogo nechetnogo = %d",numfirst);break;

 }// switch

 } while (comanda != 0); }

void in\_massiv(int mas[n])

{printf ("\nVvedite massiv:");

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

 { printf ("\n m [%d] = ", i); scanf ("%d",&mas[i]); }

}

void print\_massiv(int mas[n])

{ printf ("\nVveden massiv:");

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

 printf ("\n m [%d] = %d", i, mas[i]);

}

int summa(int mas[n])

{ int s=0;

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

 s+= mas[i];

 return s;

}

int first\_nechet(int mas[n])

{ int num=-1;

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

 if (mas[i]%2 != 0)

 {num =i; break;}

 return num;

}

int first\_nechet2(int mas[n])

{ int num =-1; int i =0;

 while (i<n && mas[i]%2 == 0) i++;

 if (i < n) num =i;

###  return num; }

### Варианты заданий к части № 1 «Одномерные массивы».

### Вариант 3

В одномерном массиве, состоящем из n целых элементов, вычислить:

1. произведение элементов массива с чётными номерами;
2. сумму элементов массива, расположенных между первым и последним нулевыми элементами.

Пример программы, выполняющей вычисления с матрицей

Задание

В матрице размером n на m вычислить:

1. Сумму всех элементов;

2. Количество строк, в которых есть числа меньше 10;

3. Количество столбцов, в которых произведение элементов является четным числом

#include "stdafx.h"

const int n = 3;

const int m = 4;

void in\_massiv(int mas[n][m]);

void print\_massiv(int mas[n][m]);

int summa(int mas[n][m]);

int chislo\_strok(int mas[n][m]);

int chislo\_stolbzov(int mas[n][m]);

void main()

{ int mas [n][m];

 int s, kolstr=-1, kolstb = -1, comanda =6;

 do {

 printf ("\n1 - vvod, 2 - print, 3 - summa, 4 - chislo strok, 5 - chislo stolbzov, 0 - out\n");

 scanf ("%d",&comanda);

 switch (comanda ){

 case 1: in\_massiv(mas); break;

 case 2: print\_massiv(mas); break;

 case 3: s=summa(mas);

 printf ("\nSumma massiva = %d", s);break;

 case 4: kolstr = chislo\_strok(mas);

 printf ("\nChislo strok = %d",kolstr);break;

 case 5: kolstb = chislo\_stolbzov(mas);

 printf ("\n Chislo stolbzov = %d",kolstb);break;

 }// switch

 } while (comanda != 0); }

void in\_massiv(int mas[n][m])

{ printf ("\nVvedite massiv:");

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

 for (int j =0; j<m; j++)

 { printf ("\n m [%d][%d] = ", i, j); scanf ("%d",&mas[i][j]);

 }}

void print\_massiv(int mas[n][m])

{ printf ("\nVveden massiv:\n");

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

 { for (int j =0; j<m; j++)

 printf ("%d ", mas[i][j]);

 printf ("\n");

 }

}

int summa(int mas[n][m])

{ int s=0;

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

 for (int j =0; j<m; j++)

 s+= mas[i][j];

 return s;

}

int chislo\_strok(int mas[n][m])

{ int num=0, kol;

 for (int i=0; i<n; i++)// цикл по строкам

 { kol =0;

 for (int j =0; j<m; j++)// цикл по столбцам

 if (mas[i][j] < 10) kol++;

 if (kol > 0)

 num++;//увеличиваем счетчик строк

 }// for i

 return num;

}

int chislo\_stolbzov(int mas[n][m])

{ int num=0, pr;

 for (int i=0; i<m; i++)// цикл по столбцам

 { pr =1;

 for (int j =0; j<n; j++)//цикл по строкам

 pr = pr \* mas[j][i];

 printf ("\n proizvedenie v stolbze % d = %d", i, pr);

 if (pr % 2 ==0)

 num++;//увеличиваем счетчик столбцов

 }// for i

 return num;

}

### Варианты заданий для части № 2 «Матрицы».

### Вариант

Дана целочисленная прямоугольная матрица. Определить количество столбцов, содержащих хотя бы один элемент, кратный 5.