Лабораторная работа №3:

Уровень 1:

№6:

using System;

using static System.Console;

using static System.Math;

namespace LaboratoryWorkNo3

{

class Program

{

static void Main(string[] args)

{

double[] vector = new double[5] { 5, 9, -4, 8, 0 };

PrintArray(vector);

ShowClarification(vector);

}

static double GetVectorLength(double[] vector)

{

double sum = 0;

for (int i = 0; i < vector.Length; i++)

sum += vector[i] \* vector[i];

double length = Sqrt(sum);

return length;

}

static void ShowClarification(double[] vector)

{

double length = GetVectorLength(vector);

Write($"L = Sqrt({vector[0]}\*{vector[0]}");

for (int i = 1; i < vector.Length; i++)

Write($" + {vector[i]}\*{vector[i]}");

WriteLine($") = {length}");

}

static void PrintArray(double[] array)

{

if (array.Length == 0)

{

WriteLine("\* Массив пустой \*");

}

else

{

WriteLine($"Массив из {array.Length} элементов:");

for (int i = 0; i < array.Length; i++)

{

WriteLine($"{i + 1}-й элемент = {array[i]}");

}

}

}

}

}

№10:

using System;

using static System.Console;

namespace L1\_task10

{

class Program

{

static void Main(string[] args)

{

double[] array = new double[10] { 3, 8, 9, 4, 0, 12, 6, 15, 29, 1 };

PrintArray(array);

Write("Введите значение P: ");

double p = double.Parse(ReadLine());

Write("Введите значение Q: ");

double q = double.Parse(ReadLine());

int count = CountOfElementsBetween(array, p, q);

WriteLine($"Кол-во элементов между {p} и {q} = {count}");

}

static int CountOfElementsBetween(double[] array, double p, double q)

{

int count = 0;

for (int i = 0; i < array.Length; i++)

{

if (array[i] > p && array[i] < q)

{

count++;

}

}

return count;

}

static void PrintArray(double[] array)

{

if (array.Length == 0)

{

WriteLine("\* Массив пустой \*");

}

else

{

WriteLine($"Массив из {array.Length} элементов:");

for (int i = 0; i < array.Length; i++)

{

WriteLine($"{i + 1}-й элемент = {array[i]}");

}

}

}

}

}

№11:

using System;

using static System.Console;

namespace L1\_task11

{

class Program

{

static void Main(string[] args)

{

double[] allNumbers = new double[10] { 3, 4, 91, -7, 0, 18, 11, -99, 19, 0 };

double[] positiveNumbers = FindPositive(allNumbers);

WriteLine("Сформированный массив из положительных чисел:");

PrintArray(positiveNumbers);

}

static double[] FindPositive(double[] numbers)

{

int count = 0;

for (int i = 0; i < numbers.Length; i++)

{

if (numbers[i] > 0)

{

count++;

}

}

double[] positiveNumbers = new double[count];

int index = -1;

for (int i = 0; i < numbers.Length; i++)

{

if (numbers[i] > 0)

{

index++;

positiveNumbers[index] = numbers[i];

}

}

return positiveNumbers;

}

static void PrintArray(double[] array)

{

if (array.Length == 0)

{

WriteLine("\* Массив пустой \*");

}

else

{

WriteLine($"Массив из {array.Length} элементов:");

for (int i = 0; i < array.Length; i++)

{

WriteLine($"{i + 1}-й элемент = {array[i]}");

}

}

}

}

}

№12:

using System;

using static System.Console;

namespace L1\_task12

{

class Program

{

static void Main(string[] args)

{

double[] numbers = new double[10] { 7, 8, -90, 25, 9, 12, -6, 11, 24, 0 };

PrintArray(numbers);

int index = GetLastNegativeIndex(numbers);

if (index == -1)

{

WriteLine("В массиве нет отрицательных чисел");

}

else

{

WriteLine("Последнее отрицательное число найдено:");

WriteLine($"Индекс = {index} (он же {index + 1}-й элемент); значение = {numbers[index]}");

}

}

static int GetLastNegativeIndex(double[] numbers)

{

for (int i = numbers.Length - 1; i >= 0; i--)

{

if (numbers[i] < 0)

{

return i;

}

}

return -1;

}

static void PrintArray(double[] array)

{

if (array.Length == 0)

{

WriteLine("\* Массив пустой \*");

}

else

{

WriteLine($"Массив из {array.Length} элементов:");

for (int i = 0; i < array.Length; i++)

{

WriteLine($"{i + 1}-й элемент = {array[i]}");

}

}

}

}

}

№13:

using System;

using static System.Console;

namespace L1\_task13

{

class Program

{

static void Main(string[] args)

{

//Индекс соответствует значению

double[] baseArray = new double[10] { 0, 1, 2, 3, 4, 5, 6, 7, 8, 9 };

double[] evenIndexArray = GetArrayWithEvenIndexes(baseArray);

double[] notEvenIndexArray = GetArrayWithNotEvenIndexes(baseArray);

WriteLine("Подмассив с чётными индексами:");

PrintArray(evenIndexArray);

WriteLine("Подмассив с нечётными индексами:");

PrintArray(notEvenIndexArray);

}

static void PrintArray(double[] array)

{

if (array.Length == 0)

{

WriteLine("\* Массив пустой \*");

}

else

{

WriteLine($"Массив из {array.Length} элементов:");

for (int i = 0; i < array.Length; i++)

{

WriteLine($"{i + 1}-й элемент = {array[i]}");

}

}

}

static double[] GetArrayWithEvenIndexes(double[] array)

{

double[] evenArray = new double[5];

for (int i = 0; i < 10; i += 2)

{

evenArray[i / 2] = array[i];

}

return evenArray;

}

static double[] GetArrayWithNotEvenIndexes(double[] array)

{

double[] notEvenArray = new double[5];

for (int i = 1; i < 10; i += 2)

{

notEvenArray[i / 2] = array[i];

}

return notEvenArray;

}

}

}

using System;

using static System.Console;

namespace L1\_task13

{

class Program

{

static void Main(string[] args)

{

//Индекс соответствует значению

double[] baseArray = new double[10] { 0, 1, 2, 3, 4, 5, 6, 7, 8, 9 };

double[] evenIndexArray = GetArrayWithEvenIndexes(baseArray);

double[] notEvenIndexArray = GetArrayWithNotEvenIndexes(baseArray);

WriteLine("Подмассив с чётными индексами:");

PrintArray(evenIndexArray);

WriteLine("Подмассив с нечётными индексами:");

PrintArray(notEvenIndexArray);

}

static void PrintArray(double[] array)

{

if (array.Length == 0)

{

WriteLine("\* Массив пустой \*");

}

else

{

WriteLine($"Массив из {array.Length} элементов:");

for (int i = 0; i < array.Length; i++)

{

WriteLine($"{i + 1}-й элемент = {array[i]}");

}

}

}

static double[] GetArrayWithEvenIndexes(double[] array)

{

double[] evenArray = new double[5];

for (int i = 0; i < 10; i += 2)

{

evenArray[i / 2] = array[i];

}

return evenArray;

}

static double[] GetArrayWithNotEvenIndexes(double[] array)

{

double[] notEvenArray = new double[5];

for (int i = 1; i < 10; i += 2)

{

notEvenArray[i / 2] = array[i];

}

return notEvenArray;

}

}

}

Уровень 2:

№5:

using System;

using static System.Console;

namespace L2\_task05

{

class Program

{

static void Main(string[] args)

{

double[] startArray = InputArray();

double[] necessaryArray = FindNegativeBetweenMinMax(startArray);

WriteLine("Массив из отриц. элементов между мин. и макс. элементами:");

PrintArray(necessaryArray);

}

static double[] InputArray()

{

Write("Введите размер массива: ");

int itemCount = int.Parse(ReadLine());

double[] array = new double[itemCount];

WriteLine($"Введите {itemCount} элементов массива:");

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

{

Write($"{i + 1}-й элемент: ");

array[i] = double.Parse(ReadLine());

}

return array;

}

static double[] FindNegativeBetweenMinMax(double[] array)

{

int minValueIndex = GetMinValueIndex(array);

int maxValueIndex = GetMaxValueIndex(array);

if (minValueIndex > maxValueIndex)

{

int temp = minValueIndex;

minValueIndex = maxValueIndex;

maxValueIndex = temp;

}

int count = 0;

for (int i = minValueIndex + 1; i < maxValueIndex; i++)

{

if (array[i] < 0)

{

count++;

}

}

double[] negativeNumbers = new double[count];

int index = -1;

for (int i = minValueIndex + 1; i < maxValueIndex; i++)

{

if (array[i] < 0)

{

index++;

negativeNumbers[index] = array[i];

}

}

return negativeNumbers;

}

static int GetMinValueIndex(double[] array)

{

int minValueIndex = 0;

double minValue = array[0];

for (int i = 0; i < array.Length; i++)

{

if (array[i] < minValue)

{

minValue = array[i];

minValueIndex = i;

}

}

return minValueIndex;

}

static int GetMaxValueIndex(double[] array)

{

int maxValueIndex = 0;

double maxValue = array[0];

for (int i = 0; i < array.Length; i++)

{

if (array[i] > maxValue)

{

maxValue = array[i];

maxValueIndex = i;

}

}

return maxValueIndex;

}

static void PrintArray(double[] array)

{

if (array.Length == 0)

{

WriteLine("\* Массив пустой \*");

}

else

{

WriteLine($"Массив из {array.Length} элементов:");

for (int i = 0; i < array.Length; i++)

{

WriteLine($"{i + 1}-й элемент = {array[i]}");

}

}

}

}

}

№6:

using System;

using static System.Console;

using static System.Math;

namespace L2\_task06

{

class Program

{

static void Main(string[] args)

{

double[] numbers = InputArray();

Write("Введите число P, которое необходимо вставить в массив: ");

double p = double.Parse(ReadLine());

double[] newNumbers = InsertNearestToAverage(numbers, p);

WriteLine("Новый массив:");

PrintArray(newNumbers);

}

static double[] InputArray()

{

Write("Введите размер массива: ");

int itemCount = int.Parse(ReadLine());

double[] array = new double[itemCount];

WriteLine($"Введите {itemCount} элементов массива:");

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

{

Write($"{i + 1}-й элемент: ");

array[i] = double.Parse(ReadLine());

}

return array;

}

static double[] InsertNearestToAverage(double[] numbers, double p)

{

double averageValue = 0;

double elementSum = 0;

for (int i = 0; i < numbers.Length; i++)

{

elementSum += numbers[i];

}

averageValue = elementSum / numbers.Length;

int nearestToAverageIndex = 0;

for (int i = 0; i < numbers.Length; i++)

{

if (Abs(numbers[i] - averageValue) < Abs(numbers[nearestToAverageIndex] - averageValue))

{

nearestToAverageIndex = i;

}

}

double[] newNumbers = new double[numbers.Length + 1];

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

{

newNumbers[i] = numbers[i];

}

//Вставка числа P после числа наиболее близкого к среднему арифметическому

newNumbers[nearestToAverageIndex + 1] = p;

for (int i = nearestToAverageIndex + 1; i < numbers.Length; i++)

{

newNumbers[i + 1] = numbers[i];

}

return newNumbers;

}

static void PrintArray(double[] array)

{

if (array.Length == 0)

{

WriteLine("\* Массив пустой \*");

}

else

{

WriteLine($"Массив из {array.Length} элементов:");

for (int i = 0; i < array.Length; i++)

{

WriteLine($"{i + 1}-й элемент = {array[i]}");

}

}

}

}

}

№9:

using System;

using static System.Console;

namespace L2\_task09

{

class Program

{

static void Main(string[] args)

{

double[] numbers = InputArray();

double averageBetweenMinMax = FindAverageBetweenMinMax(numbers);

WriteLine($"Среднее арифметическое между мин. и макс. элементами массива = {averageBetweenMinMax}");

}

static double[] InputArray()

{

Write("Введите размер массива: ");

int itemCount = int.Parse(ReadLine());

double[] array = new double[itemCount];

WriteLine($"Введите {itemCount} элементов массива:");

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

{

Write($"{i + 1}-й элемент: ");

array[i] = double.Parse(ReadLine());

}

return array;

}

static void PrintArray(double[] array)

{

if (array.Length == 0)

{

WriteLine("\* Массив пустой \*");

}

else

{

WriteLine($"Массив из {array.Length} элементов:");

for (int i = 0; i < array.Length; i++)

{

WriteLine($"{i + 1}-й элемент = {array[i]}");

}

}

}

static double FindAverageBetweenMinMax(double[] array)

{

int minValueIndex = GetMinValueIndex(array);

int maxValueIndex = GetMaxValueIndex(array);

if (minValueIndex > maxValueIndex)

{

int temp = minValueIndex;

minValueIndex = maxValueIndex;

maxValueIndex = temp;

}

int count = 0;

double elementSum = 0;

for (int i = minValueIndex + 1; i < maxValueIndex; i++)

{

count++;

elementSum += array[i];

}

double average = elementSum / count;

return average;

}

static int GetMinValueIndex(double[] array)

{

int minValueIndex = 0;

double minValue = array[0];

for (int i = 0; i < array.Length; i++)

{

if (array[i] < minValue)

{

minValue = array[i];

minValueIndex = i;

}

}

return minValueIndex;

}

static int GetMaxValueIndex(double[] array)

{

int maxValueIndex = 0;

double maxValue = array[0];

for (int i = 0; i < array.Length; i++)

{

if (array[i] > maxValue)

{

maxValue = array[i];

maxValueIndex = i;

}

}

return maxValueIndex;

}

}

}

№10:

using System;

using static System.Console;

namespace L2\_task10

{

class Program

{

static void Main(string[] args)

{

double[] numbers = InputArray();

double[] necessaryNumbers = EraseMinPositive(numbers);

WriteLine("Массив без минимального положительного элемента:");

PrintArray(necessaryNumbers);

}

static double[] EraseMinPositive(double[] array)

{

int positiveNumberIndex = -1;

for (int i = 0; i < array.Length; i++)

{

if (array[i] > 0)

{

positiveNumberIndex = i;

break;

}

}

if (positiveNumberIndex == -1)

{

WriteLine("Положительных элементов нет");

return array;

}

int minPositiveIndex = positiveNumberIndex;

for (int i = minPositiveIndex; i < array.Length; i++)

{

if ((array[i] > 0) && (array[i] < array[minPositiveIndex]))

{

minPositiveIndex = i;

}

}

double[] newArray = new double[array.Length - 1];

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

{

newArray[i] = array[i];

}

for (int i = minPositiveIndex + 1; i < array.Length; i++)

{

newArray[i - 1] = array[i];

}

return newArray;

}

static double[] InputArray()

{

Write("Введите размер массива: ");

int itemCount = int.Parse(ReadLine());

double[] array = new double[itemCount];

WriteLine($"Введите {itemCount} элементов массива:");

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

{

Write($"{i + 1}-й элемент: ");

array[i] = double.Parse(ReadLine());

}

return array;

}

static void PrintArray(double[] array)

{

if (array.Length == 0)

{

WriteLine("\* Массив пустой \*");

}

else

{

WriteLine($"Массив из {array.Length} элементов:");

for (int i = 0; i < array.Length; i++)

{

WriteLine($"{i + 1}-й элемент = {array[i]}");

}

}

}

}

}

№11:

using System;

using static System.Console;

namespace L2\_task11

{

class Program

{

static void Main(string[] args)

{

double[] array = InputArray();

Write("Введите число P, которое необходимо вставить в массив: ");

double p = double.Parse(ReadLine());

double[] newArray = InsertAfterLastPositive(array, p);

WriteLine("Новый массив:");

PrintArray(newArray);

}

static double[] InputArray()

{

Write("Введите размер массива: ");

int itemCount = int.Parse(ReadLine());

double[] array = new double[itemCount];

WriteLine($"Введите {itemCount} элементов массива:");

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

{

Write($"{i + 1}-й элемент: ");

array[i] = double.Parse(ReadLine());

}

return array;

}

static void PrintArray(double[] array)

{

if (array.Length == 0)

{

WriteLine("\* Массив пустой \*");

}

else

{

WriteLine($"Массив из {array.Length} элементов:");

for (int i = 0; i < array.Length; i++)

{

WriteLine($"{i + 1}-й элемент = {array[i]}");

}

}

}

static double[] InsertAfterLastPositive(double[] array, double p)

{

int lastPositiveIndex = GetLastPositiveIndex(array);

if (lastPositiveIndex == -1)

{

WriteLine("В массиве нет положительных чисел");

return array;

}

double[] biggerArray = new double[array.Length + 1];

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

{

biggerArray[i] = array[i];

}

biggerArray[lastPositiveIndex + 1] = p;

for (int i = lastPositiveIndex + 1; i < array.Length; i++)

{

biggerArray[i + 1] = array[i];

}

return biggerArray;

}

static int GetLastPositiveIndex(double[] array)

{

int index = -1;

for (int i = array.Length - 1; i > 0; i--)

{

if (array[i] > 0)

{

index = i;

break;

}

}

return index;

}

}

}

№13:

using System;

using static System.Console;

namespace L2\_task13

{

class Program

{

static void Main(string[] args)

{

double[] array = InputArray();

double[] swappedArray = SwapMaxEvenAndIndex(array);

WriteLine("Новый массив:");

PrintArray(swappedArray);

}

static double[] SwapMaxEvenAndIndex(double[] array)

{

int maxIndex = 0;

for (int i = 0; i < array.Length; i += 2)

{

if (array[i] > array[maxIndex])

{

maxIndex = i;

}

}

WriteLine($"Замена A[{maxIndex}] = {array[maxIndex]} На индекс = {maxIndex}");

array[maxIndex] = maxIndex;

return array;

}

static double[] InputArray()

{

Write("Введите размер массива: ");

int itemCount = int.Parse(ReadLine());

double[] array = new double[itemCount];

WriteLine($"Введите {itemCount} элементов массива:");

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

{

Write($"{i + 1}-й элемент: ");

array[i] = double.Parse(ReadLine());

}

return array;

}

static void PrintArray(double[] array)

{

if (array.Length == 0)

{

WriteLine("\* Массив пустой \*");

}

else

{

WriteLine($"Массив из {array.Length} элементов:");

for (int i = 0; i < array.Length; i++)

{

WriteLine($"{i + 1}-й элемент = {array[i]}");

}

}

}

}

}

№15:

using System;

using static System.Console;

namespace L2\_task15

{

class Program

{

static void Main(string[] args)

{

Write("Массив A, ");

double[] a = InputArray();

Write("Массив B, ");

double[] b = InputArray();

Write("Введите K (индекс вставки массива B в A): ");

int k = int.Parse(ReadLine());

double[] c = InsertArrayIntoArray(a, b, k);

WriteLine("Склеенный массив:");

PrintArray(c);

}

static double[] InputArray()

{

Write("Введите размер массива: ");

int itemCount = int.Parse(ReadLine());

double[] array = new double[itemCount];

WriteLine($"Введите {itemCount} элементов массива:");

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

{

Write($"элемент [{i}]: ");

array[i] = double.Parse(ReadLine());

}

return array;

}

static void PrintArray(double[] array)

{

if (array.Length == 0)

{

WriteLine("\* Массив пустой \*");

}

else

{

WriteLine($"Массив из {array.Length} элементов:");

for (int i = 0; i < array.Length; i++)

{

WriteLine($"[{i}] = {array[i]}");

}

}

}

static double[] InsertArrayIntoArray(double[] baseArray, double[] insertingArray, int k)

{

double[] bigArray = new double[baseArray.Length + insertingArray.Length];

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

{

bigArray[i] = baseArray[i];

}

for (int i = 0; i < insertingArray.Length; i++)

{

bigArray[k + 1 + i] = insertingArray[i];

}

for (int i = k + 1; i < baseArray.Length; i++)

{

int index = i + insertingArray.Length;

bigArray[index] = baseArray[i];

}

return bigArray;

}

}

}

Уровень 3:

№1:

using System;

using static System.Console;

namespace L3\_task01

{

class Program

{

static void Main(string[] args)

{

double[] array = InputArray();

double[] maxValueIndexes = GetMaxValueIndexArray(array);

WriteLine("Массив индексов с максимальными значениями массива:");

PrintArray(maxValueIndexes);

}

static double[] InputArray()

{

Write("Введите размер массива: ");

int itemCount = int.Parse(ReadLine());

double[] array = new double[itemCount];

WriteLine($"Введите {itemCount} элементов массива:");

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

{

Write($"элемент [{i}]: ");

array[i] = double.Parse(ReadLine());

}

return array;

}

static void PrintArray(double[] array)

{

if (array.Length == 0)

{

WriteLine("\* Массив пустой \*");

}

else

{

WriteLine($"Массив из {array.Length} элементов:");

for (int i = 0; i < array.Length; i++)

{

WriteLine($"[{i}] = {array[i]}");

}

}

}

static double[] GetMaxValueIndexArray(double[] array)

{

if (array.Length == 0)

{

return array;

}

int maxValueIndex = 0;

int maxValueCount = 1;

for (int i = 0; i < array.Length; i++)

{

if (array[i] == array[maxValueIndex])

{

maxValueCount++;

}

if (array[i] > array[maxValueIndex])

{

maxValueIndex = i;

maxValueCount = 1;

}

}

double[] maxValueIndexes = new double[maxValueCount];

int index = -1;

for (int i = 0; i < array.Length; i++)

{

if (array[i] == array[maxValueIndex])

{

index++;

maxValueIndexes[index] = i;

}

}

return maxValueIndexes;

}

}

}

№5:

using System;

using static System.Console;

namespace L3\_task05

{

class Program

{

static void Main(string[] args)

{

double[] array = InputArray();

double[] sortedArray = SortEvenIndexes(array);

WriteLine("Массив с упорядоченными чётными индексами:");

PrintArray(sortedArray);

}

static double[] SortEvenIndexes(double[] array)

{

for (int i = 0; i < array.Length; i += 2)

{

for (int j = 0; j < array.Length - 2; j++)

{

if (((array.Length - j) >= 2) && (array[j] > array[j + 2]))

{

double temp = array[j + 2];

array[j + 2] = array[j];

array[j] = temp;

}

}

}

return array;

}

static double[] InputArray()

{

Write("Введите размер массива: ");

int itemCount = int.Parse(ReadLine());

double[] array = new double[itemCount];

WriteLine($"Введите {itemCount} элементов массива:");

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

{

Write($"элемент [{i}]: ");

array[i] = double.Parse(ReadLine());

}

return array;

}

static void PrintArray(double[] array)

{

if (array.Length == 0)

{

WriteLine("\* Массив пустой \*");

}

else

{

WriteLine($"Массив из {array.Length} элементов:");

for (int i = 0; i < array.Length; i++)

{

WriteLine($"[{i}] = {array[i]}");

}

}

}

}

}

№8:

using System;

using static System.Console;

namespace L3\_task08

{

class Program

{

static void Main(string[] args)

{

double[] array = InputArray();

double[] sortedArray = SortNegativeNumbers(array);

WriteLine("Массив с упорядоченными отрицательными элементами:");

PrintArray(sortedArray);

}

static double[] SortNegativeNumbers(double[] array)

{

int negativeNumberCount = 0;

for (int i = 0; i < array.Length; i++)

{

if (array[i] < 0)

{

negativeNumberCount++;

}

}

if (negativeNumberCount == 0)

{

WriteLine("В массиве нет отрицательных чисел");

return array;

}

int[] negativeNumberIndexes = new int[negativeNumberCount];

int index = -1;

for (int i = 0; i < array.Length; i++)

{

if (array[i] < 0)

{

index++;

negativeNumberIndexes[index] = i;

}

}

for (int i = 0; i < negativeNumberIndexes.Length; i += 1)

{

for (int j = 0; j < negativeNumberIndexes.Length - 1; j++)

{

if (array[negativeNumberIndexes[j]] > array[negativeNumberIndexes[j + 1]])

{

double temp = array[negativeNumberIndexes[j + 1]];

array[negativeNumberIndexes[j + 1]] = array[negativeNumberIndexes[j]];

array[negativeNumberIndexes[j]] = temp;

}

}

}

return array;

}

static double[] InputArray()

{

Write("Введите размер массива: ");

int itemCount = int.Parse(ReadLine());

double[] array = new double[itemCount];

WriteLine($"Введите {itemCount} элементов массива:");

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

{

Write($"элемент [{i}]: ");

array[i] = double.Parse(ReadLine());

}

return array;

}

static void PrintArray(double[] array)

{

if (array.Length == 0)

{

WriteLine("\* Массив пустой \*");

}

else

{

WriteLine($"Массив из {array.Length} элементов:");

for (int i = 0; i < array.Length; i++)

{

WriteLine($"[{i}] = {array[i]}");

}

}

}

}

}

№9:

using System;

using static System.Console;

using static System.Math;

namespace L3\_task09

{

class Program

{

static void Main(string[] args)

{

double[] array = InputArray();

int orderLength = FindLongestOrder(array);

WriteLine($"Длина самой длинной упорядоченной последовательности = {orderLength}");

}

static int FindLongestOrder(double[] array)

{

int orderLength = 1, maxOrderLength = 1;

for (int i = 1; i < array.Length; i++)

{

if (array[i] >= array[i - 1])

{

orderLength++;

}

else

{

maxOrderLength = Max(orderLength, maxOrderLength);

orderLength = 1;

}

}

maxOrderLength = Max(orderLength, maxOrderLength);

orderLength = 1;

for (int i = 1; i < array.Length; i++)

{

if (array[i] <= array[i - 1])

{

orderLength++;

}

else

{

maxOrderLength = Max(orderLength, maxOrderLength);

orderLength = 1;

}

}

maxOrderLength = Max(orderLength, maxOrderLength);

return maxOrderLength;

}

static double[] InputArray()

{

Write("Введите размер массива: ");

int itemCount = int.Parse(ReadLine());

double[] array = new double[itemCount];

WriteLine($"Введите {itemCount} элементов массива:");

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

{

Write($"элемент [{i}]: ");

array[i] = double.Parse(ReadLine());

}

return array;

}

}

}

№12:

using System;

using static System.Console;

namespace L3\_task12

{

class Program

{

static void Main(string[] args)

{

double[] array = InputArray();

double[] withoutNegativeArray = DeleteNegativeNumbers(array);

Write("Массив без отрицательных элементов:");

PrintArray(withoutNegativeArray);

}

static double[] DeleteNegativeNumbers(double[] array)

{

int negativeCount = 0;

for (int i = 0; i < array.Length; i++)

{

if (array[i] < 0)

{

negativeCount++;

}

}

if (negativeCount == 0)

{

WriteLine("В массиве нет отрицательных чисел");

return array;

}

double[] withoutNegative = new double[array.Length - negativeCount];

int index = -1;

for (int i = 0; i < array.Length; i++)

{

if (array[i] >= 0)

{

index++;

withoutNegative[index] = array[i];

}

}

return withoutNegative;

}

static double[] InputArray()

{

Write("Введите размер массива: ");

int itemCount = int.Parse(ReadLine());

double[] array = new double[itemCount];

WriteLine($"Введите {itemCount} элементов массива:");

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

{

Write($"элемент [{i}]: ");

array[i] = double.Parse(ReadLine());

}

return array;

}

static void PrintArray(double[] array)

{

if (array.Length == 0)

{

WriteLine("\* Массив пустой \*");

}

else

{

WriteLine($"Массив из {array.Length} элементов:");

for (int i = 0; i < array.Length; i++)

{

WriteLine($"[{i}] = {array[i]}");

}

}

}

}

}

№13:

using System;

using static System.Console;

namespace L3\_task13

{

class Program

{

static void Main(string[] args)

{

double[] array = InputArray();

double[] withoutDuplicates = DeleteDuplicates(array);

WriteLine("Массив без повторяющихся элементов:");

PrintArray(withoutDuplicates);

}

static double[] DeleteDuplicates(double[] array)

{

int differentItemsCount = 0;

for (int i = 0; i < array.Length; i++)

{

bool hasDuplicates = false;

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

{

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

{

hasDuplicates = true;

break;

}

}

if (!hasDuplicates)

{

differentItemsCount++;

}

}

double[] uniqueItems = new double[differentItemsCount];

int index = -1;

for (int i = 0; i < array.Length; i++)

{

bool hasDuplicates = false;

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

{

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

{

hasDuplicates = true;

break;

}

}

if (!hasDuplicates)

{

index++;

uniqueItems[index] = array[i];

}

}

return uniqueItems;

}

static double[] InputArray()

{

Write("Введите размер массива: ");

int itemCount = int.Parse(ReadLine());

double[] array = new double[itemCount];

WriteLine($"Введите {itemCount} элементов массива:");

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

{

Write($"элемент [{i}]: ");

array[i] = double.Parse(ReadLine());

}

return array;

}

static void PrintArray(double[] array)

{

if (array.Length == 0)

{

WriteLine("\* Массив пустой \*");

}

else

{

WriteLine($"Массив из {array.Length} элементов:");

for (int i = 0; i < array.Length; i++)

{

WriteLine($"[{i}] = {array[i]}");

}

}

}

}

}