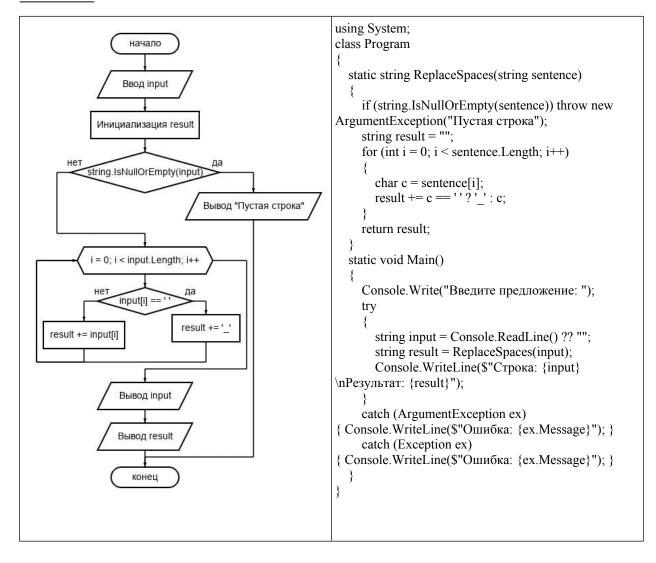
ПУСТОЙ ЛИСТ ДЛЯ ЗАГОТОВОК

Содержание

Задача А 1/1 (Зачтено)	3
Задача А 2/1 (Зачтено)	5
Блок с пояснением Задачи А 2/2	7
Задача А 3/2 (Зачтено)	8
Задача А 4/3 (Зачтено)	10
Задача А 5/3 (Зачтено)	12
Задача А 6/3 (Зачтено)	14
Задача А 7/3 (Зачтено)	16
Задача А 8/4 (Зачтено)	18
Задача А 9/4 (Зачтено)	20
Задача А 10/5 (Зачтено)	22
Задача А 11/5 (Зачтено)	24
Задача А 12/5 ИСПРАВЛЕНО (заменена блок схема)	26
Задача А 13/7 (Зачтено)	28
Задача В 1/1 (Зачтено)	30
Задача В 2/1 (Зачтено)	32
Задача В 3/2 (Зачтено)	34
Задача В 4/4	36
Задача В 5/4	38
Задача В 6/4	40
20daug P 7/6	42

Задача А 12/5 ИСПРАВЛЕНО (заменена блок схема)

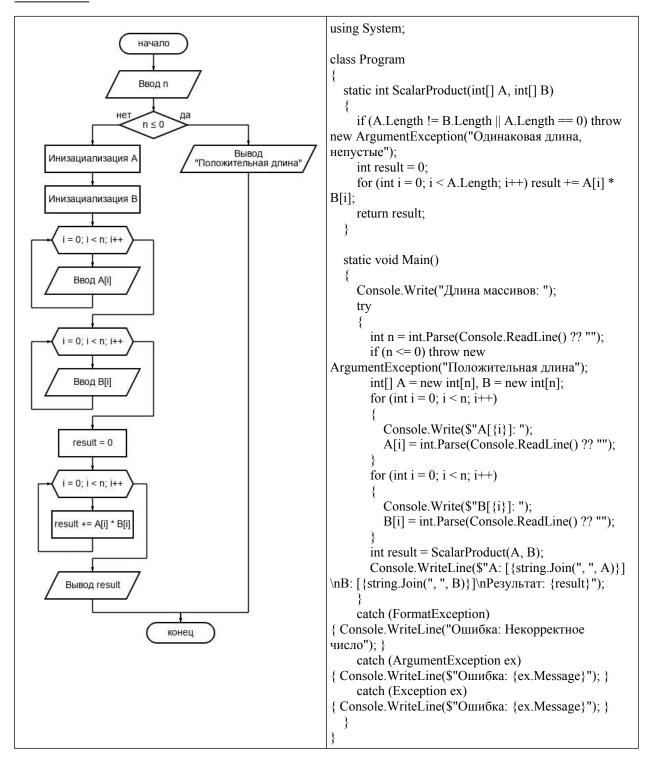
Дано предложение. Все пробелы в нем заменить на символ « ».



```
≡ ТЕСТЫ для тестов ^-^ ≡
--- Тест: Позитивный тест (предложение с пробелами) ---
Входная строка: Hello world test
Результат: Hello_world_test
Г Прошёл
--- Тест: Негативный тест (без пробелов) ---
Входная строка: NoSpacesHere
Результат: NoSpacesHere
Г Прошёл
--- Тест: Пограничный тест (один пробел) ---
Входная строка:
Результат: _
Г Прошёл
--- Тест: Исключительный тест (пустая строка) ---
Прошёл (исключение: Слово не может быть пустым)
====== Закончили =======
Введите предложение: Hello World No Space
Входная строка: Hello World No Space
Результат: Hello_World_No_Space
```

Задача В 4/4

Даны два одномерных массива А и В. Найти их скалярное произведение.



```
≡ ТЕСТЫ для тестов ^-^ ≡
--- Тест: Позитивный тест (3 элемента) ---
Массив А: [1, 2, 3]
Массив В: [4, 5, 6]
Результат: 32
Г Прошёл
--- Тест: Негативный тест (нулевой массив) ---
Массив А: [0, 0, 0]
Массив В: [1, 2, 3]
Результат: 0
Прошёл
--- Тест: Пограничный тест (1 элемент) ---
Массив А: [1]
Массив В: [2]
Результат: 2
Грошёл
--- Тест: Исключительный тест (разные длины) ---
Массив А: [1, 2]
Массив В: [1]
🔽 Прошёл (исключение: Массивы должны быть одинаковой длины и непустыми)
====== Закончили ======
Введите длину массивов: 3
Введите элементы массива А:
A[0]: 56
A[1]: 94
A[2]: 10
Введите элементы массива В:
B[0]: 34
B[1]: 67
B[2]: 1
Массив А: [56, 94, 10]
Массив В: [34, 67, 1]
Результат: 8212
```

Задача В 5/4

В соревнованиях по прыжкам в длину участвовали N-спортсменов. Каждый спортсмен мог сделать M-попыток. Результаты (длины прыжков) занесены в таблицу A[M][N]. Если результат не защитывали, то вместо длины прыжков заносили ноль. Определить, какой по номеру спортсмен имеет лучший результат в 4-ой попытке.

```
using System;
                                                             class Program
                             Ввод N
                                                               static int FindBestAthlete(int[,] A, int M, int N)
                             Ввод М
                                                                  if (M < 4 \parallel N \le 0 \parallel A.GetLength(0) != M \parallel
                                                             A.GetLength(1) != N)
                                                                    throw new ArgumentException("Некорректные
                                                             размеры или M<4");
                                    Вывод
"Положительные размеры"
                                                                  int maxJump = A[3, 0], best = 0;
                                                                  for (int j = 1; j < N; j++)
Инициализация А
                                                                     if (A[3, j] > maxJump)
 i = 0: i < M: i++
                                                                       maxJump = A[3, j];
                                                                       best = j;
  = 0; j < N; j++
                                                                  return best + 1;
   Ввод А[і, ј]
                                                               static void Main()
maxJump = A[3, 0]
                                                                  Console.Write("Попытки М: ");
                                                                  Console. Write("Спортсмены N: ");
   best = 0
                                                                    int M = int.Parse(Console.ReadLine() ?? "");
  = 1; j < N; j++
                                                                    int N = int.Parse(Console.ReadLine()??"");
                                                                    if (M \le 0 \parallel N \le 0) throw new
A[3, j] > maxJump
                                                             ArgumentException("Положительные размеры");
                                                                    int[,] A = new int[M, N];
              maxJump = A[3, j]
                                                                     for (int i = 0; i < M; i++)
                                                                       for (int j = 0; j < N; j++)
                  best = j
                                                                          Console. Write(\{i+1\},\{j+1\}\}: ");
                                                                          A[i, j] = int.Parse(Console.ReadLine()??"");
                                                                     int result = FindBestAthlete(A, M, N);
  Вывод best
                                                                    Console.WriteLine("Таблица:");
                                                                    for (int i = 0; i < M; i++)
                             начало
                                                                       int[] row = new int[N];
                                                                       for (int j = 0; j < N; j++) row[j] = A[i, j];
                                                                       Console.WriteLine($"[{string.Join(", ", row)}]");
                                                                    Console.WriteLine($"Результат: {result}");
                                                                  catch (FormatException) { Console.WriteLine("Ошибка:
                                                             Некорректное число"); }
                                                                  catch (ArgumentException ex)
                                                             { Console.WriteLine($"Ошибка: {ex.Message}"); }
                                                                  catch (Exception ex) { Console.WriteLine($"Ошибка:
                                                             {ex.Message}"); }
```

```
≡ ТЕСТЫ для тестов ^-^ ≡
--- Тест: Позитивный тест (4-я попытка, max=12) ---
M: 5, N: 3
Таблица:
[1, 2, 3]
[4, 5, 6]
[7, 8, 9]
[10, 12, 11]
[13, 14, 15]
Результат: 2
Прошёл
--- Тест: Негативный тест (все нули в 4-й попытке) ---
M: 5, N: 3
Таблица:
[1, 2, 3]
[4, 5, 6]
[7, 8, 9]
[0, 0, 0]
[10, 11, 12]
Результат: 1
🔽 Прошёл
--- Тест: Пограничный тест (1 спортсмен) ---
M: 5, N: 1
Таблица:
[1]
[2]
[3]
[4]
[5]
Результат: 1
🔽 Прошёл
--- Тест: Исключительный тест (М<4) ---
🔽 Прошёл (исключение: Некорректные размеры таблицы или недостаточно попыток)
======= Закончили =======
Введите количество попыток М: 4
Введите количество спортсменов N: 4
Введите результаты прыжков:
A[1,1]: 1
A[1,2]: 2
A[1,3]: 3
A[1,4]: 9
A[2,1]: 8
A[2,2]: 7
A[2,3]: 10
A[2,4]: 12
A[3,1]: 13
A[3,2]: 14
A[3,3]: 1
A[3,4]: 67
A[4,1]: 32
A[4,2]: 65
A[4,3]: 87
A[4,4]: 44
Таблица:
[1, 2, 3, 9]
[8, 7, 10, 12]
[13, 14, 1, 67]
[32, 65, 87, 44]
Результат: 3
```

Задача В 6/4

Дан двумерный массив размером 5 * 8, заполненный случайным образом. Вставить перед всеми строками, первый элемент которых делится на 3, строку из нулей.

```
using System;
    начало
                                                 class Program
 ициализация модуля rand
                                                    static int[,] InsertZeroRows(int[,] A)
    i = 0; i < 5; i++
                                                       if (A.GetLength(0) != 5 || A.GetLength(1) != 8) throw
                                                 new ArgumentException("Размер 5х8");
    j = 0; j < 8; j++
                                                       int rowsToInsert = 0;
 A[i, j] = rand.Next(1, 101)
                                                       for (int i = 0; i < 5; i++)
                                                          if (A[i, 0] \% 3 == 0) rowsToInsert++;
                                                       int[,] result = new int[5 + rowsToInsert, 8];
    i = 0; i < 5; i++
                                                       int currentRow = 0;
                                                       for (int i = 0; i < 5; i++)
    j = 0; j < 8; j++
                                                          if (A[i, 0] \% 3 == 0)
     Вывод А[і, ј]
                                                            for (int j = 0; j < 8; j++) result[currentRow, j] = 0;
   rowsToInsert = 0
                                                            currentRow++;
                                                          for (int j = 0; j < 8; j++) result[currentRow, j] = A[i, j];
                                                         currentRow++;
             rowsToInsert++
                                                       return result;
 Инициализация result
                                                    static void Main()
    currentRow = 0
                                                       int[,] A = new int[5, 8];
                                                       Random rand = new Random();
                                                       for (int i = 0; i < 5; i++)
                                                          for (int j = 0; j < 8; j++)
                                                            A[i, j] = rand.Next(1, 101);
             j = 0; j < 8; j++
                                                          Console.WriteLine("Массив:");
                                                          for (int i = 0; i < 5; i++)
                                                            Console.WriteLine($"[{string.Join(", ",
                                                 Enumerable.Range(0, 8).Select(j \Rightarrow A[i, j]))}]");
                                                          int[,] result = InsertZeroRows(A);
    j = 0; j < 8; j++
                                                          Console.WriteLine("Результат:");
result[currentRow, j] = A[i,j]
                                                          for (int i = 0; i < result.GetLength(0); i++)
                                                            Console.WriteLine($"[{string.Join(", "
                                                 Enumerable.Range(0, 8).Select(j => result[i, j]))}]");
    currentRow++
                                                       catch (ArgumentException ex)
                                                 { Console.WriteLine($"Ошибка: {ex.Message}"); }
= 0; i < result.GetLength(0); i++
                                                       catch (Exception ex) { Console.WriteLine($"Ошибка:
    j = 0; j < 8; j++
                                                 {ex.Message}"); }
    .
Вывод result[i, j]
```

```
≡ ТЕСТЫ для тестов ^-^ ≡
--- Тест: Позитивный тест (3 строки с первым элементом, делящимся на 3) ---
Массив:
[3, 1, 2, 3, 4, 5, 6, 7]
[1, 2, 3, 4, 5, 6, 7, 8]
[6, 1, 2, 3, 4, 5, 6, 7]
[2, 2, 3, 4, 5, 6, 7, 8]
[9, 1, 2, 3, 4, 5, 6, 7]
Результат:
Массив:
[0, 0, 0, 0, 0, 0, 0, 0]
[3, 1, 2, 3, 4, 5, 6, 7]
[1, 2, 3, 4, 5, 6, 7, 8]
[0, 0, 0, 0, 0, 0, 0, 0]
[6, 1, 2, 3, 4, 5, 6, 7]
[2, 2, 3, 4, 5, 6, 7, 8]
[0, 0, 0, 0, 0, 0, 0, 0]
[9, 1, 2, 3, 4, 5, 6, 7]
🔽 Прошёл
--- Тест: Негативный тест (нет строк с первым элементом, делящимся на 3) ---
Массив:
[1, 1, 2, 3, 4, 5, 6, 7]
[2, 2, 3, 4, 5, 6, 7, 8]
[4, 1, 2, 3, 4, 5, 6, 7]
[5, 2, 3, 4, 5, 6, 7, 8]
[7, 1, 2, 3, 4, 5, 6, 7]
Результат:
Массив:
[1, 1, 2, 3, 4, 5, 6, 7]
[2, 2, 3, 4, 5, 6, 7, 8]
[4, 1, 2, 3, 4, 5, 6, 7]
[5, 2, 3, 4, 5, 6, 7, 8]
[7, 1, 2, 3, 4, 5, 6, 7]
Прошёл
--- Тест: Пограничный тест (первый элемент 0, не вставляем строку) ---
Массив:
[0, 1, 2, 3, 4, 5, 6, 7]
[1, 2, 3, 4, 5, 6, 7, 8]
[2, 1, 2, 3, 4, 5, 6, 7]
[4, 2, 3, 4, 5, 6, 7, 8]
[5, 1, 2, 3, 4, 5, 6, 7]
Результат:
Массив:
[0, 0, 0, 0, 0, 0, 0, 0]
[0, 1, 2, 3, 4, 5, 6, 7]
[1, 2, 3, 4, 5, 6, 7, 8]
[2, 1, 2, 3, 4, 5, 6, 7]
[4, 2, 3, 4, 5, 6, 7, 8]
[5, 1, 2, 3, 4, 5, 6, 7]
🔽 Прошёл
--- Тест: Исключительный тест (неверный размер) ---
🔽 Прошёл (исключение: Массив должен быть размером 5х8)
======= Закончили =======
Заполнение массива случайными числами (1-100):
Массив:
[47, 62, 22, 5, 88, 93, 45, 25]
[97, 87, 72, 47, 1, 21, 85, 33]
[94, 69, 11, 90, 40, 68, 100, 78]
[76, 35, 10, 82, 49, 46, 49, 84]
[20, 89, 66, 89, 77, 43, 36, 47]
Результат:
Массив:
[47, 62, 22, 5, 88, 93, 45, 25]
[97, 87, 72, 47, 1, 21, 85, 33]
[94, 69, 11, 90, 40, 68, 100, 78]
[76, 35, 10, 82, 49, 46, 49, 84]
[20, 89, 66, 89, 77, 43, 36, 47]
```

Задача В 7/6

Удалить значение P в упорядоченном по неубыванию массиве X(1) < = X(2) < ... < = X(n).

```
using System;
                                      class Program
                                        static int[] RemoveValue(int[] X, int P)
                                           if (X.Length == 0) throw new ArgumentException("Пустой
                                      массив");
                                           for (int i = 1; i < X.Length; i++)
                                             if (X[i] \le X[i-1]) throw new
                                      ArgumentException("Неупорядоченный массив");
                                           int left = 0, right = X.Length - 1, leftBound = -1, rightBound = -1;
                                           while (left <= right)
                                             int mid = left + (right - left) / 2;
                                             if (X[mid] == P) { leftBound = mid; right = mid - 1; }
                                             else if (X[mid] < P) left = mid + 1;
                                             else right = mid - 1;
                                           left = 0; right = X.Length - 1;
                                           while (left <= right)
                                             int mid = left + (right - left) / 2;
                                             if (X[mid] == P) { rightBound = mid; left = mid + 1; }
                                             else if (X[mid] < P) left = mid + 1;
mid = left + (right - left) / 2
                                             else right = mid - 1;
                                           if (leftBound == -1) return (int[])X.Clone();
                                           int[] result = new int[X.Length - (rightBound - leftBound + 1)];
                                           int index = 0;
                                           for (int i = 0; i < leftBound; i++) result[index++] = X[i];
                                           for (int i = rightBound + 1; i < X.Length; i++) result[index++] =
                                      X[i];
                                           return result;
                                        static void Main()
                                           Console.Write("Длина массива: ");
                                           Console.Write("Значение Р: ");
                                             int n = int.Parse(Console.ReadLine() ?? "");
                                             int P = int.Parse(Console.ReadLine() ?? "");
                                             if (n <= 0) throw new ArgumentException("Положительная
                                      длина");
                                             int[] X = new int[n];
for (int i = 0; i < n; i++)
                                                Console.Write($"X[{i}]: ");
                                                X[i] = int.Parse(Console.ReadLine()??"");
                                             int[] result = RemoveValue(X, P);
                                      Console.WriteLine($"X: [{string.Join(", ", X)}]\nP: {P} \nPeзультат: [{string.Join(", ", result)}]");
                                           catch (FormatException) { Console.WriteLine("Ошибка:
                                      Некорректное число"); }
                                           catch (ArgumentException ex) { Console.WriteLine($"Ошибка:
                                      {ex.Message}"); }
начало
                                           catch (Exception ex) { Console.WriteLine($"Ошибка:
                                      {ex.Message}"); }
                                        }
```