Пермский Национальный Исследовательский Политехнический Университет

Электротехнический факультет

Кафедра «Информационные технологии и автоматизированные системы»

Лабораторная работа №4

по дисциплине

«Объектно-ориентированное программирование»

Выполнил:

студент группы ИВТ-19-2б

Шеретов Марк Алексеевич

Проверила:

доцент кафедры ИТАС

Викентьева О. Л.

Пермь 2020

**Содержание**

1. Постановка задачи 3

2. Анализ задачи 4

3. Проектирование 8

4. Блок-схемы функций 10

5. Программа на C# 25

6. Тестирование 38

1. **Постановка задачи**

1) Сформировать массив из n элементов с помощью датчика случайных чисел (n задается пользователем с клавиатуры).

2) Распечатать массив.

3) Выполнить удаление указанных элементов из массива.

4) Выполнить добавление указанных элементов в массив.

5) Выполнить перестановку элементов в массиве.

6) Выполнить поиск указанных в массиве элементов и подсчитать количество сравнений, необходимых для поиска нужного элемента.

7) Выполнить сортировку массива указанным методом.

8) Выполнить поиск указанных элементов в отсортированном массиве и подсчитать количество сравнений, необходимых для поиска нужного элемента.

**Вариант 19**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Удаление** | **Добавление** | **Перестановка** | **Поиск** | **Сортировка** |
| Максимальный элемент | N элементов, начиная с номера K | Чётные элементы переставить в начало массива, нечётные — в конец | Элемент с заданным ключом (значением) | Простой обмен |

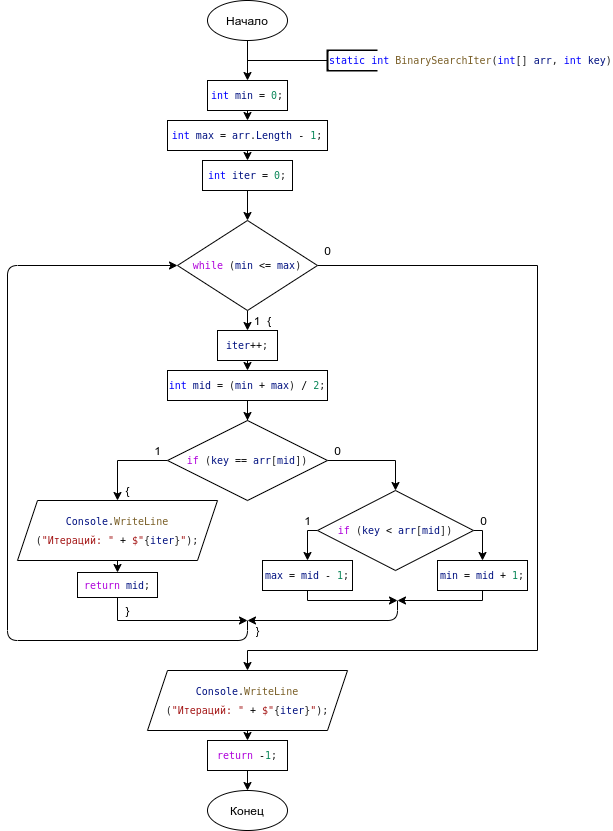
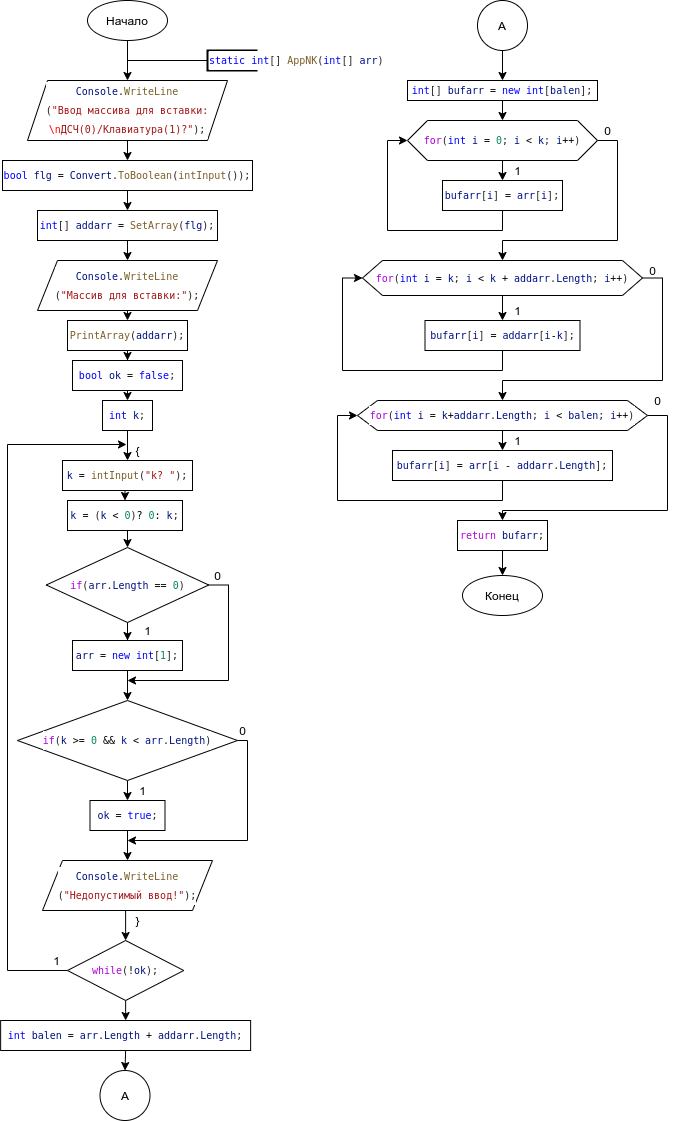
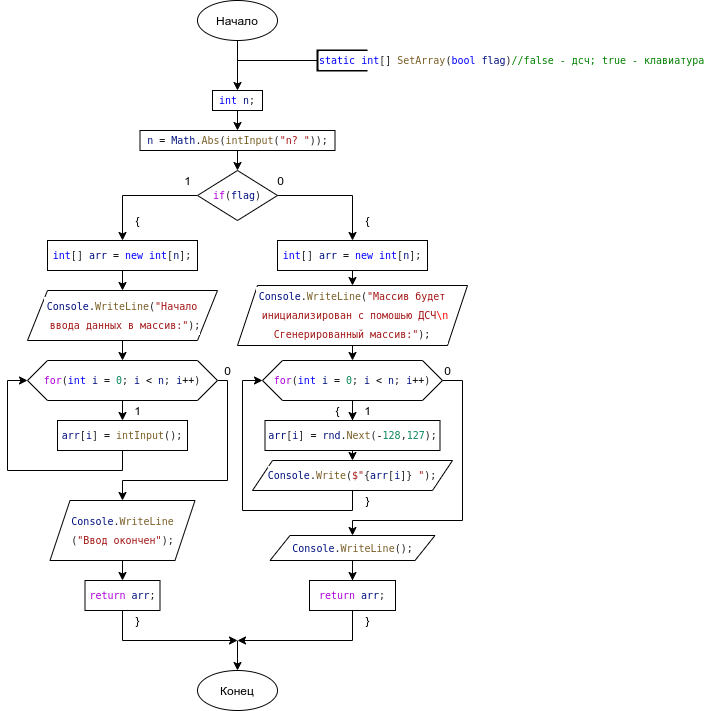
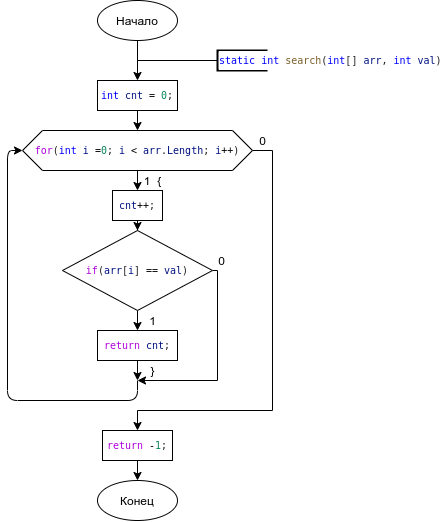
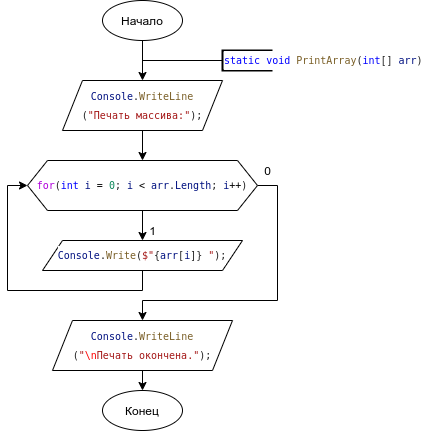
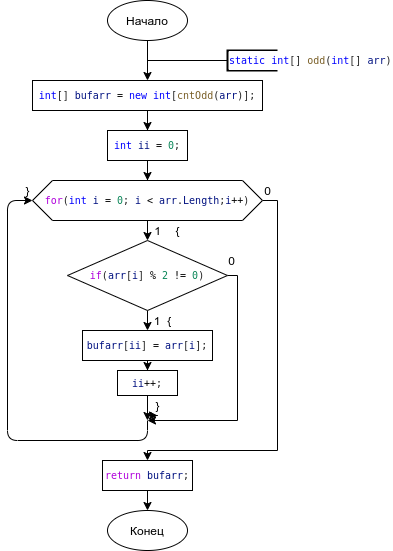
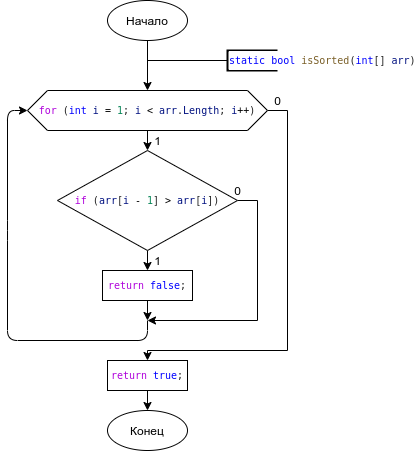
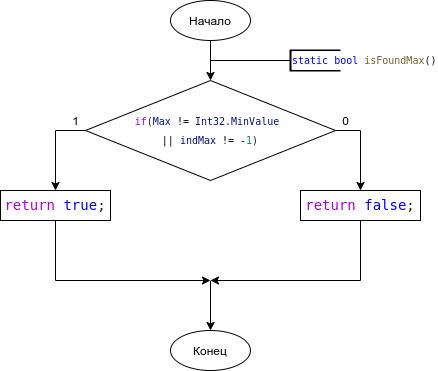
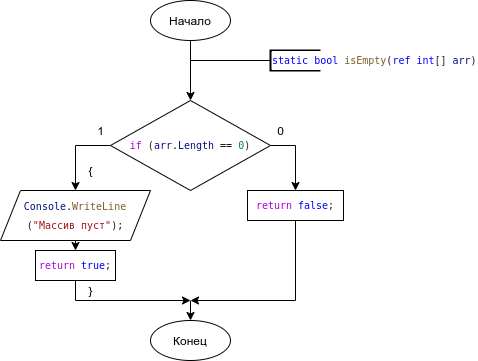
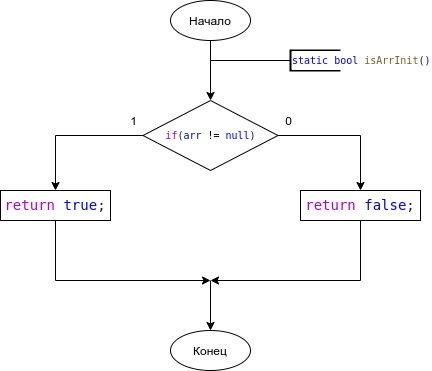
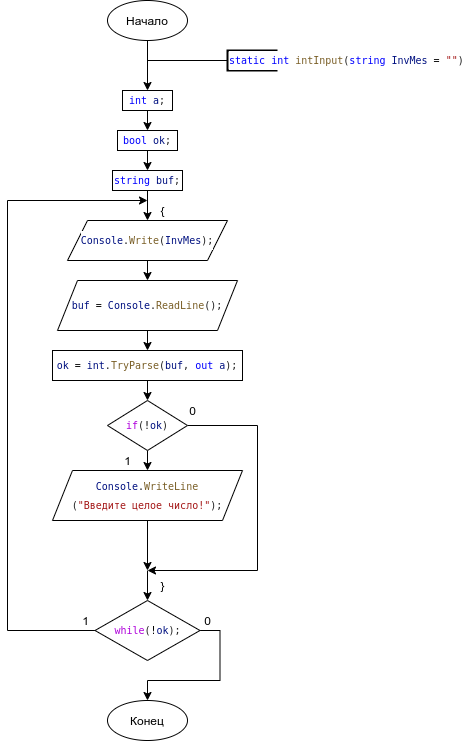
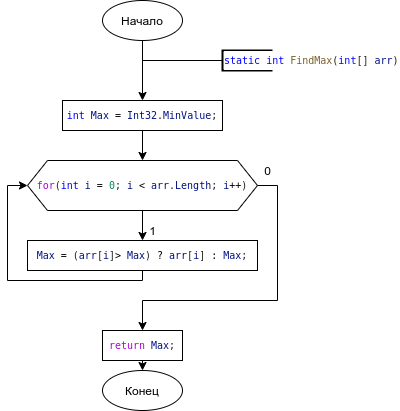
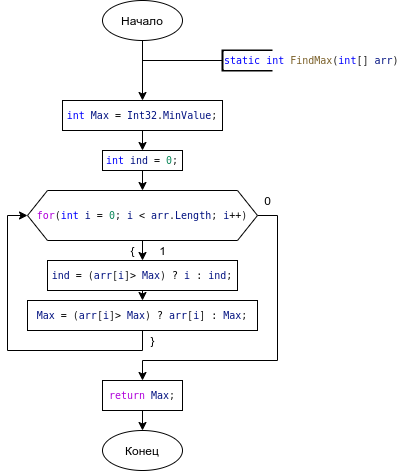
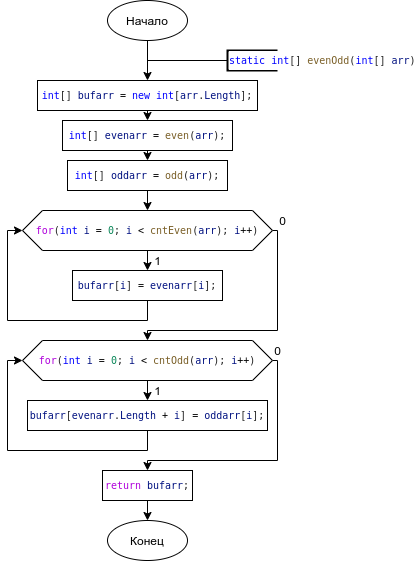
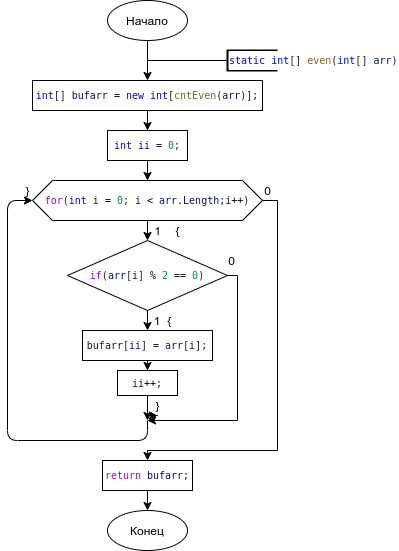
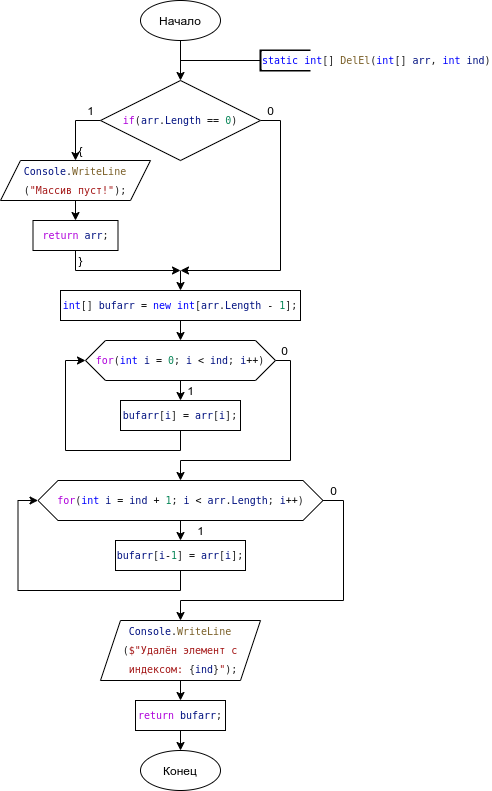
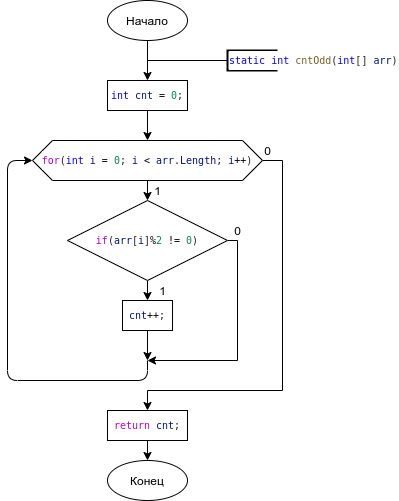
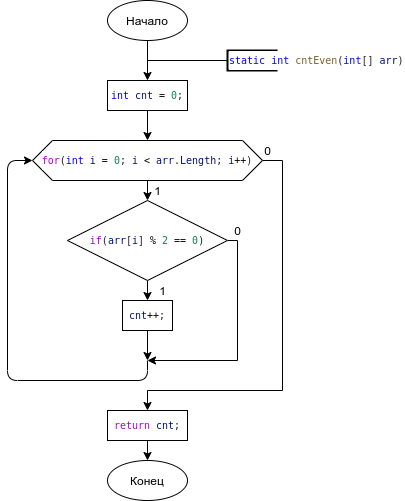
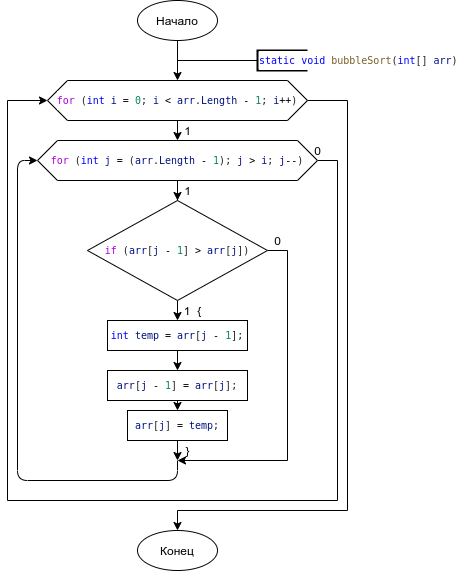
**2. Анализ задачи**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **№** | **Описание функции** | **Входные данные** | **Классы входных данных** | **Выходные данные** |
| **1** | Формирование массива | Флаг  Количество элементов массива N | Флаг != 0  Целое число N > 0 | Массив, заданный в ручную |
| Флаг = 0  Целое число N > 0 | Массив, заданный при помощи ДСЧ |
| - Флаг != 0  - Целое число N <= 0  - Вещественное число  - Строка | Сообщение об ошибке, повторный ввод |
| - Флаг = 0  - Целое число N <= 0  - Вещественное число  - Строка | Сообщение об ошибке, повторный ввод |
| **3** | Печать массива | Массив | Массив непустой | Массив напечатан |
| - Массив пустой не выд. память под массив | Сообщение об ошибке |
| **4** | Удаление максимальных элементов | Массив | Массив непустой | Удаление максимального элемента  Измененный массив  Измененный размер массив |
| - Массив пустой не выд. память под массив | Сообщение об ошибке  Массив не меняется  Размер массива не меняется |
| **5** | Добавление N элементов после K элемента с помощью ДСЧ | - Количество элементов массива N  - Массив  - индекс K | - Целое число 0<=k <длинна массива  - Массив непустой | Добавление элементов в массив  Измененный массив  Измененный размер массива |
| - Массив пустой или не выд. память под массив | Сообщение об ошибке |
| N - вещественное число, символ, строка, <0 | Сообщение об ошибке, повторный ввод |
| K – вещественное число, символ, строка, <0, > размера массива. |
| **6** | Добавление K элементов в начало массива вручную | - Количество элементов массива K  - Массив  - Элементы массива | - Целое число 0<=K< длина массива.  - Массив непустой  - Элементы – целые числа | Добавление элементов в массив  Измененный массив  Измененный размер массива |
| - Массив пустой или не выд. память под массив | Сообщение об ошибке |
| K – вещественное число, символ, строка, <0, >длина массива | Сообщение об ошибке, повторный ввод |
| N - вещественное число, символ, строка, <0 |
| Элементы – вещественное число, символ |
| **7** | Перестановка чётных и нечётных элементов | Массив  Массив чётных  Массив нечётных | Массив непустой | Массив меняется  Размер массива не меняется |
| Массив пустой или не выд. память под массив | Сообщение об ошибке |
| **8** | Сортировка простым обменом (пузырьком) | Массив | - Массив непустой  - Массив не отсортирован | Сортировка массива  Массив меняется  Размер массива не меняется |
| - Массив пустой или не выд. память под массив | Сообщение об ошибке |
| Массив отсортирован | Массив не меняется  Размер массива не меняется |
| **9** | Бинарный поиск целого числа в массиве | Массив  Ключ | Массив отсортирован  Массив непустой  Ключ – целое число | Индекс найденного числа или -1, если число не нашлось  Массив не меняется  Размер массива не меняется |
| Массив не отсортирован  Массив непустой  Ключ – целое число | Сортировка массива  Индекс найденного числа или -1, если число не нашлось  Массив меняется  Размер массива не меняется |
| Ключ – не целое число | Сообщение об ошибке, повторный ввод |
| - Массив пустой или не выд. память под массив | Сообщение об ошибке |
| **Вспомогательные функции** | | | | |
| **10** | Ввод целого числа с клавиатуры | Строка | Строка – представляет собой целое число | Число |
| Не является целым числом | Сообщение об ошибке, повторный ввод |
| Очень большое целое число (переполнение) |
| **11** | Ввод числа элементов массива N | Количество элементов в массиве | Целое число N > 0 | Число N |
| Целое число N <= 0 | Сообщение об ошибке, повторный ввод |
| Символ/строка |
| Веществ. число |
| **12** | Ввод числа добавляемых элементов массива K | Количество элементов | Целое число K >= 0 | Число K |
| Целое число K < 0 | Сообщение об ошибке, повторный ввод |
| Символ/строка |
| Веществ. число |
| **13** | Поиск максимума в массиве | Массив |  | Печать максимума |
| **14** | Проверка на отсортированность | Массив | Массив не отсортирован | false |
| Массив отсортирован | true |
| **15** | Проверка на выделение памяти для массива | Массив | Память выделена | false |
| Память не выделена | true |
| **16** | Проверка на пустоту массива | Массив | Массив непустой | false |
| Массив пустой | true |
| **17** | Печать диалога для ввода команд пользователя |  |  |  |
| **18** | Создание массива чётных элементов | Массив | Массив непустой  Есть чётные элементы | Массив чётных элементов из исходного массива |
| Массив непустой  Нет чётных элементов | Пустой массив |
| Массив пустой | Пустой массив |
| **19** | Создание массива нечётных элементов | Массив | Массив непустой  Есть нечётные элементы | Массив нечётных элементов из исходного массива |
| Массив непустой  Нет нечётных элементов | Пустой массив |
| Массив пустой | Пустой массив |

1. **Проектирование**

|  |  |  |  |
| --- | --- | --- | --- |
| **№** | **Функция** | **Входные данные** | **Выходные данные** |
| **Основные функции** | | | |
| **1** | SetArray (Формирование массива с помощью ДСЧ) | bool flag – флаг, указывающий на ввод способ ввода данных в массив. | int[] arr – массив |
| **2** | PrintArray(Печать массива) | int[] arr – ссылка на массив | Печать массива  Сообщение об ошибке |
| **3** | DelEl (Удаление элемента по индексу) | int[] arr – ссылка на массив  int ind – индекс элемента | int[] arr – массив  Сообщение об ошибке |
| **4** | AppNK (Добавление N элементов начиная с K элемента) | int[] arr – ссылка на массив  k – кол-во добавляемых элементов массива | int[] arr – массив  Сообщение об ошибке |
| **5** | evenOdd (Постановка) | int[] arr – ссылка на массив | int[] arr – массив  Сообщение об ошибке |
| **6** | FindIndMax (Поиск индекса максимального элемента) | int[] arr – ссылка на массив | int ind – индекс максимума  Нет четных чисел  Сообщение об ошибке |
| **7** | bubbleSort (Сортировка простым обменом (пузырьком)) | int[] arr – ссылка на массив | int[] arr – массив  Сообщение массив отсортирован  Сообщение об ошибке |
| **8** | BinarySearchIter (бинарный поиск) | int[] arr – ссылка на массив  int key – ключ, целое число | Число найдено, индекс  Число не найдено  Сообщение об ошибке |
| **Вспомогательные функции** | | | |
| **9** | intInput(Ввод целого числа с клавиатуры) | Строка, которую вводит пользователь  String invMsg – сообщение-приглашение | Целое число  Сообщение об ошибке |
| **10** | isSorted (Проверка на отсортированность) | ref int[] arr – ссылка на массив | True  false |
| **11** | isEmpty (Проверка на пустоту массива) | ref int[] arr – ссылка на массив | True  false |
| **12** | PrintMainMenu (Печать диалога для ввода команд пользователя) |  |  |

**4 Блок-схемы функций**



**5 Программа на языке C#**

using System;

//Вариант лабораторной работы №19.

namespace OOP.Practice4

{

    class Program

    {   //Переменные

        static int[] arr; //Массив

        static int menuX = 1;

        static Random rnd = new Random(setRndSeed());

        static int Max = Int32.MinValue;

        static int indMax = -1;

        //Функции и методы

        static bool isFoundMax()

        {

            if(Max != Int32.MinValue || indMax != -1)

                return true;

            else return false;

        }

         static bool isArrInit()

        {

            if(arr != null) return true;

            else return false;

        }

        static int intInput(string InvMes = "")//ввод любой переменной int.

        {

            int a;

            bool ok;

            string buf;

            do

            {

                Console.Write(InvMes);

                buf = Console.ReadLine();

                ok = int.TryParse(buf, out a);

                if(!ok) Console.WriteLine("Введите целое число!");

            }while(!ok);

            return a;

        }

        static int[] SetArray(bool flag)//false - дсч; true - клавиатура

        {

            int n;

            n = Math.Abs(intInput("n? "));

            if(flag)

            {

                int[] arr = new int[n];

                Console.WriteLine("Начало ввода данных в массив:");

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

                {

                    arr[i] = intInput();

                }

                Console.WriteLine("Ввод окончен");

                return arr;

            }

            else

            {

                int[] arr = new int[n];

                Console.WriteLine("Массив будет инициализирован с помошью ДСЧ\nСгенерированный массив:");

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

                {

                    arr[i] = rnd.Next(-128,127);

                    Console.Write($"{arr[i]} ");

                }

                Console.WriteLine();

                return arr;

            }

        }

        static void PrintArray(int[] arr)//вывод массива в консоль

        {

            Console.WriteLine("Печать массива:");

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

            {

                Console.Write($"{arr[i]} ");

            }

            Console.WriteLine("\nПечать окончена.");

        }

        static int FindMax(int[] arr)//поиск максимального значения

        {

            int Max = Int32.MinValue;

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

            {

                Max = (arr[i]> Max) ? arr[i] : Max;

            }

            return Max;

        }

        static int FindIndMax(int[] arr)//поиск индекс максимального значения

        {

            int Max = Int32.MinValue;

            int ind = 0;

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

            {

                ind = (arr[i]> Max) ? i : ind;

                Max = (arr[i]> Max) ? arr[i] : Max;

            }

            return ind;

        }

        static int[] DelEl(int[] arr, int ind)//Удаление элемента массива по индексу

        {

            if(arr.Length == 0)

            {

                Console.WriteLine("Массив пуст!");

                return arr;

            }

            int[] bufarr = new int[arr.Length - 1];

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

            {

                bufarr[i] = arr[i];

            }

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

            {

                bufarr[i-1] = arr[i];

            }

            Console.WriteLine($"Удалён элемент с индексом: {ind}");

            return bufarr;

        }

        static int[] AppNK(int[] arr)

        {

            Console.WriteLine("Ввод массива для вставки:\nДСЧ(0)/Клавиатура(1)?");

            bool flg = Convert.ToBoolean(intInput());

            int[] addarr = SetArray(flg);

            Console.WriteLine("Массив для вставки:");

            PrintArray(addarr);

            bool ok = false;

            int k;

            do

            {

                k = intInput("k? ");

                k = (k < 0)? 0: k;

                if(arr.Length == 0)

                    arr = new int[1];

                if(k >= 0 && k <= arr.Length)

                {

                    ok = true;

                }

                Console.WriteLine("Недопустимый ввод!");

            }while(!ok);

            int balen = arr.Length + addarr.Length; //длина bufarr

            int[] bufarr = new int[balen];

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

            {

                bufarr[i] = arr[i];

            }

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

            {

                bufarr[i] = addarr[i-k];

            }

            for(int i = k+addarr.Length; i < balen; i++)

            {

                bufarr[i] = arr[i - addarr.Length];

            }

            return bufarr;

        }

        static void printMainMenu(int x)

        {

            switch (x)

            {

                case 1:

                    Console.WriteLine("1)Создать массив \*\n2)Найти максимальный элемент\n3)Удалить максимальный элемент\n4)Вставить подмассив\n5)Четные - начало / Нечётные - конец\n6)Поиск\n7)Сортировка простым обменом\n8)Вывод массива\n9)Бинарный поиск");

                    break;

                case 2:

                    Console.WriteLine("1)Создать массив\n2)Найти максимальный элемент \*\n3)Удалить максимальный элемент\n4)Вставить подмассив\n5)Четные - начало / Нечётные - конец\n6)Поиск\n7)Сортировка простым обменом\n8)Вывод массива\n9)Бинарный поиск");

                    break;

                case 3:

                    Console.WriteLine("1)Создать массив\n2)Найти максимальный элемент\n3)Удалить максимальный элемент \*\n4)Вставить подмассив\n5)Четные - начало / Нечётные - конец\n6)Поиск\n7)Сортировка простым обменом\n8)Вывод массива\n9)Бинарный поиск");

                    break;

                case 4:

                    Console.WriteLine("1)Создать массив\n2)Найти максимальный элемент\n3)Удалить максимальный элемент\n4)Вставить подмассив \*\n5)Четные - начало / Нечётные - конец\n6)Поиск\n7)Сортировка простым обменом\n8)Вывод массива\n9)Бинарный поиск");

                    break;

                case 5:

                    Console.WriteLine("1)Создать массив\n2)Найти максимальный элемент\n3)Удалить максимальный элемент\n4)Вставить подмассив\n5)Четные - начало / Нечётные - конец \*\n6)Поиск\n7)Сортировка простым обменом\n8)Вывод массива\n9)Бинарный поиск");

                    break;

                case 6:

                    Console.WriteLine("1)Создать массив\n2)Найти максимальный элемент\n3)Удалить максимальный элемент\n4)Вставить подмассив\n5)Четные - начало / Нечётные - конец\n6)Поиск \*\n7)Сортировка простым обменом\n8)Вывод массива\n9)Бинарный поиск");

                    break;

                case 7:

                    Console.WriteLine("1)Создать массив\n2)Найти максимальный элемент\n3)Удалить максимальный элемент\n4)Вставить подмассив\n5)Четные - начало / Нечётные - конец\n6)Поиск\n7)Сортировка простым обменом \*\n8)Вывод массива\n9)Бинарный поиск");

                    break;

                case 8:

                    Console.WriteLine("1)Создать массив\n2)Найти максимальный элемент\n3)Удалить максимальный элемент\n4)Вставить подмассив\n5)Четные - начало / Нечётные - конец\n6)Поиск\n7)Сортировка простым обменом\n8)Вывод массива \*\n9)Бинарный поиск");

                    break;

                case 9:

                    Console.WriteLine("1)Создать массив\n2)Найти максимальный элемент\n3)Удалить максимальный элемент\n4)Вставить подмассив\n5)Четные - начало / Нечётные - конец\n6)Поиск\n7)Сортировка простым обменом\n8)Вывод массива\n9)Бинарный поиск \*");

                    break;

                default:

                    Console.WriteLine("1)Создать массив\n2)Найти максимальный элемент\n3)Удалить максимальный элемент\n4)Вставить подмассив\n5)Четные - начало / Нечётные - конец\n6)Поиск\n7)Сортировка простым обменом\n8)Вывод массива\n9)Бинарный поиск");

                    break;

            }

        }

        static void callFunc(int x)

        {

            Console.Clear();

            switch(x)

            {

                case 1:

                    Console.WriteLine("Ввод: ДСЧ(0)/Клавиатура(1)?");

                    bool flag = Convert.ToBoolean(intInput());

                    arr = SetArray(flag);

                    PrintArray(arr);

                    Console.ReadKey();

                    break;

                case 2:

                    if(isArrInit())

                    {

                        if(arr.Length == 0)

                        {

                            Console.WriteLine("Массив пуст!\nНевозможно найти максимум!");

                            Console.ReadKey();

                        }

                        else

                        {

                            Max = FindMax(arr);

                            indMax = FindIndMax(arr);

                            Console.WriteLine($"Max = {Max}\nindMax = {indMax}");

                            Console.ReadKey();

                        }

                    }

                    else

                    {

                        Console.WriteLine("Ошибка!\nМассив ещё не инициализирован!");

                        Console.ReadKey();

                    }

                    break;

                case 3:

                    if(isArrInit() && isFoundMax())

                    {

                        indMax = FindIndMax(arr);

                        arr = DelEl(arr,indMax);

                        Console.ReadKey();

                    }

                    else

                    {

                        Console.WriteLine("Ошибка!\nЛибо не инициализирован массив\nЛибо ещё не найден максимальный элемент\nЛибо пуст");

                        Console.ReadKey();

                    }

                    break;

                case 4:

                    if(isArrInit())

                    {

                        arr = AppNK(arr);

                        PrintArray(arr);

                    }

                    else

                    {

                        Console.WriteLine("Ошибка!\nНе инициализирован массив");

                    }

                    break;

                case 5:

                    if(isArrInit())

                    {

                        arr = evenOdd(arr);

                        PrintArray(arr);

                        Console.ReadKey();

                    }

                    else Console.WriteLine("Ошибка!\nНе инициализирован массив");

                    break;

                case 6:

                    if(isArrInit())

                    {

                        int val = intInput("Введите искомое число: ");

                        int rezsearch = search(arr, val);

                        if(rezsearch > -1) Console.WriteLine($"Искомое значение val = {val} найдено!\nИтераций для поиска = {rezsearch}");

                        else Console.WriteLine("Заданное значение в массиве не представленно");

                        Console.ReadKey();

                    }

                    else Console.WriteLine("Ошибка!\nНе инициализирован массив");

                    break;

                case 7:

                    if(isArrInit())

                    {

                        bubbleSort(arr);

                        PrintArray(arr);

                        Console.ReadKey();

                    }

                    else Console.WriteLine("Ошибка!\nНе инициализирован массив");

                    break;

                case 8:

                    if(isArrInit())

                    {

                        PrintArray(arr);

                        Console.ReadKey();

                    }

                    else

                    {

                        Console.WriteLine("Ошибка!\nНе инициализирован массив");

                        Console.ReadKey();

                    }

                    break;

                case 9:

                    if(isArrInit() && isSorted(arr))

                    {

                        int key = intInput("Введите ключ: ");

                        int a = BinarySearchIter(arr,key);

                        int rezsearch = search(arr,key) - 1;

                        if(a == -1) Console.WriteLine("Ключа в массиве не оказалось!");

                        else

                        {

                            Console.WriteLine("Ключевое значение найдено!");

                            Console.WriteLine($"Индекс элемента: {rezsearch}");

                        }

                        Console.ReadKey();

                    }

                    else

                    {

                        Console.WriteLine("Ошибка!\nЛибо не инициализирован массив\nЛибо массив не отсортирован!");

                        Console.ReadKey();

                    }

                    break;

                default:

                    Console.WriteLine("Ошибка!");

                    break;

            }

        }

        static void eventMenu()

        {

            ConsoleKeyInfo key;

            do

            {

                key = Console.ReadKey();

                //Выбор пунктов меню на кнопки

                if(key.Key == ConsoleKey.D1) callFunc(1);

                if(key.Key == ConsoleKey.D2) callFunc(2);

                if(key.Key == ConsoleKey.D3) callFunc(3);

                if(key.Key == ConsoleKey.D4) callFunc(4);

                if(key.Key == ConsoleKey.D5) callFunc(5);

                if(key.Key == ConsoleKey.D6) callFunc(6);

                if(key.Key == ConsoleKey.D7) callFunc(7);

                if(key.Key == ConsoleKey.D8) callFunc(8);

                if(key.Key == ConsoleKey.D9) callFunc(9);

                if(key.Key == ConsoleKey.UpArrow) menuX--;      //обработка нажатия стрелки вверх

                if(key.Key == ConsoleKey.DownArrow) menuX++;    //Обработка нажатия стрелки вниз

                if(menuX < 1) menuX = 9; //Перескок

                if(menuX > 9) menuX = 1; //Перескок

                if(key.Key == ConsoleKey.Enter) callFunc(menuX);

                Console.Clear();

                printMainMenu(menuX);

            }while(key.Key != ConsoleKey.Escape);

        }

        static int cntEven(int[] arr)

        {

            int cnt = 0;

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

            {

                if(arr[i] % 2 == 0)

                    cnt++;

            }

            return cnt;

        }

        static int[] even(int[] arr)

        {

            int[] bufarr = new int[cntEven(arr)];

            int ii = 0;

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

            {

                if(arr[i] % 2 == 0)

                {

                    bufarr[ii] = arr[i];

                    ii++;

                }

            }

            return bufarr;

        }

        static int cntOdd(int[] arr)

        {

            int cnt = 0;

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

            {

                if(arr[i]%2 != 0)

                    cnt++;

            }

            return cnt;

        }

        static int[] odd(int[] arr)

        {

            int[] bufarr = new int[cntOdd(arr)];

            int ii = 0;

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

            {

                if(arr[i] % 2 != 0)

                {

                    bufarr[ii] = arr[i];

                    ii++;

                }

            }

            return bufarr;

        }

        static int[] evenOdd(int[] arr)

        {

            int[] bufarr = new int[arr.Length];

            int[] evenarr = even(arr);

            int[] oddarr = odd(arr);

            for(int i = 0; i < cntEven(arr); i++)

            {

                bufarr[i] = evenarr[i];

            }

            for(int i = 0; i < cntOdd(arr); i++)

            {

                bufarr[evenarr.Length + i] = oddarr[i];

            }

            return bufarr;

        }

        static void bubbleSort(int[] arr)

        {

             // Для всех элементов

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

            {

                for (int j = (arr.Length - 1); j > i; j--) // для всех элементов после i-ого

                {

                    if (arr[j - 1] > arr[j]) // если текущий элемент меньше предыдущего

                    {

                        int temp = arr[j - 1]; // меняем их местами

                        arr[j - 1] = arr[j];

                        arr[j] = temp;

                    }

                }

            }

        }

        static bool isEmpty(ref int[] arr) //проверка на пустоту

        {

            if (arr.Length == 0)

            {

                Console.WriteLine("Массив пуст");

                return true;

            }

            else return false;

        }

        static bool isSorted(int[] arr)

        {

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

            {

                if (arr[i - 1] > arr[i])

                {

                    return false;

                }

            }

            return true;

        }

        static int BinarySearchIter(int[] arr, int key) //бинарный поиск целого числа в массиве

        {

            int min = 0;

            int max = arr.Length - 1;

            int iter = 0;

            while (min <= max)

            {

                iter++;

                int mid = (min + max) / 2;

                if (key == arr[mid])

                {

                    Console.WriteLine("Итераций: " + $"{iter}");

                    return mid;

                }

                else if (key < arr[mid]) max = mid - 1;

                else min = mid + 1;

            }

            Console.WriteLine("Итераций: " + $"{iter}");

            return -1;

        }

        static int search(int[] arr, int val)

        {

            int cnt = 0;

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

            {

                cnt++;

                if(arr[i] == val)

                    return cnt;

            }

            return -1;

        }

        static int setRndSeed()

        {

            return intInput("Введите зерно для генерации псевдослучайных чисел: ");

        }

        static void Main(string[] args)

        {

            Console.Clear();

            printMainMenu(-1);

            eventMenu();

        }

    }

}

**6 Тестирование**

**1. Формирование массива с помощью ДСЧ**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Номер теста** | **Входные данные** | **Ожидаемый результат** | **Выходные данные** | **Примечание** |
| **1** | 5 | 5 чисел | 114 29 1 98 -107 |  |
| **2** | 0 | Ошибка | Введите целое число > 0 |  |
| **3** | -1 | Ошибка | Введите целое число > 0 |  |
| **4** | 1 | 1 число | 56 |  |
| **5** | asd | Ошибка | Неверное число |  |
| **6** | 33333333333333 | Ошибка | Неверное число |  |
| **7** | 1,234 | Ошибка | Неверное число |  |

**2. Формирование массива с клавиатуры**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Номер теста** | **Входные данные** | **Ожидаемый результат** | **Выходные данные** | **Примечание** |
| **1** | N = 5  1 2 3 4 5 | 5 чисел | 1 2 3 4 5 |  |
| **2** | N = 1  5 | 1 число | 5 |  |
| **3** | N = asd | Ошибка | Неверное число |  |
| **4** | N = 6  1 2 a 4 5 6 | Ошибка | Неверное число, запрос повторного ввода после a |  |
| **5** | N = 6  1 2 1,2 4 5 6 | Ошибка | Неверное число, запрос повторного ввода после 1,2 |  |
| **6** | N = 0 | Ошибка | Введите целое число > 0 |  |
| **7** | N = -1 | Ошибка | Введите целое число > 0 |  |

**3. Печать массива**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Номер теста** | **Входные данные** | **Ожидаемый результат** | **Выходные данные** | **Примечание** |
| **1** | arr = 5 2 3 1 6 | 5 чисел | 5 2 3 1 6 |  |
| **2** | arr = [] | Ошибка | Массив пуст |  |
| **3** | arr = null | Ошибка | Ошибка! Не инициализирован массив |  |

**4. Удаление максимума**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Номер теста** | **Входные данные** | **Ожидаемый результат** | **Выходные данные** | **Примечание** |
| **1** | 5 2 1 3 222 3 | 5 чисел | 5 2 1 3 3 |  |
| **2** | 1 | Пустой массив | Пустой массив |  |
| **3** | 5 5 5 5 5 5 | 5 чисел | 5 5 5 5 5 |  |
| **4** | -225 5 | 1 число | -225 |  |
| **5** | arr = [] | Ошибка | Массив пуст |  |
| **6** | arr = null | Ошибка | Создайте массив |  |

**5. Добавление N элементов начиная с K элемента массива с клавиатуры**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Номер теста** | **Входные данные** | **Ожидаемый результат** | **Выходные данные** | **Примечание** |
| **1** | arr = 5 2 3 5  n = 4  k = 1  1 1 2 2 | 8 чисел | 5 1 1 2 2 2 3 5 |  |
| **2** | arr = 1  n = 2  k = 1  2 3 | 3 числа | 1 2 3 |  |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **3** | arr = 7 6 5  n = 0  k = 0 | 3 числа | 7 6 5 |  |
| **4** | arr = 7 6 5  n = -5 | Ошибка | Введите целое неотрицательное число |  |
| **5** | arr = [] | Ошибка | Массив пуст |  |
| **6** | arr = null | Ошибка | Создайте массив |  |

**6. Добавление K элементов в начало массива с помощью ДСЧ**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Номер теста** | **Входные данные** | **Ожидаемый результат** | **Выходные данные** | **Примечание** |
| **1** | arr = 5 2 3 5  n = 4  k = 0 | 8 чисел | 23 -1 33 54 5 2 3 5 |  |
| **2** | arr = 1  n = 2  k = 1 | 3 числа | 1 3 -2 |  |
| **3** | arr = 7 6 5  n = 0  k = 0 | 3 числа | 7 6 5 |  |
| **4** | arr = 7 6 5  k = -5 | Ошибка | Введите целое неотрицательное число |  |
| **5** | arr = [] | Ошибка | Массив пуст |  |
| **6** | arr = null | Ошибка | Создайте массив |  |

**7. Чётные в начало, нечётные – в конец**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Номер теста** | **Входные данные** | **Ожидаемый результат** | **Выходные данные** | **Примечание** |
| **1** | arr = 1 2 3 4 | 2 4 1 3 | 2 4 1 3 |  |
| **2** | arr = 1 | 1 | 1 |  |
| **3** | arr = -2 5 -25 | -2 5 -25 | -2 5 -25 |  |
| **4** | arr = [] | Ошибка | Массив пуст |  |
| **5** | arr = null | Ошибка | Создайте массив |  |

**8. Сортировка пузырьком**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Номер теста** | **Входные данные** | **Ожидаемый результат** | **Выходные данные** | **Примечание** |
| **1** | arr = -2 3 -5 1 | -2 -5 1 3 | -2 -5 1 3 |  |
| **2** | arr = 1 | Массив уже отсортирован | Массив уже отсортирован |  |
| **3** | arr = -2 -1 5 36 7 | Массив уже отсортирован | Массив уже отсортирован |  |
| **4** | arr = 3 1 2 1 1 3 | 1 1 1 2 3 3 | 1 1 1 2 3 3 |  |
| **5** | arr = [] | Ошибка | Массив пуст |  |
| **6** | arr = null | Ошибка | Создайте массив |  |

**9. Бинарный поиск**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Номер теста** | **Входные данные** | **Ожидаемый результат** | **Выходные данные** | **Примечание** |
| **1** | arr = -2 3 -5 1  key = 3 | -2 -5 1 3  Индекс: 3 | -2 -5 1 3  Итераций: 3  Индекс: 3 | +Сортировка |
| **2** | arr = -2 3 -5 1  key = -1 | -2 -5 1 3  Число не найдено | -2 -5 1 3  Итераций: 2  Заданное число не найдено | +Сортировка |
| **3** | arr = 225  key = 225 | 225  Индекс: 0 | 225  Итераций: 0  Индекс: 0 | Уже отсортировано |
| **4** | arr = 225  key = 2 | 225  Число не найдено | 225  Итераций: 1  Заданное число не найдено | Уже отсортировано |
| **5** | arr = -2 1 1 3  key = -2 | -2 1 1 3  Индекс: 0 | -2 1 1 3  Итераций: 2  Индекс: 0 | Уже отсортировано |
| **6** | arr = -2 1 1 3  key = 1 | -2 1 1 3  Индекс: 1 | 2 1 1 3  Итераций: 1  Индекс: 1 | Уже отсортировано |
| **7** | arr = -2 1 1 3  key = 3 | -2 1 1 3  Индекс: 3 | 2 1 1 3  Итераций: 3  Индекс: 3 | Уже отсортировано |
| **8** | arr = [] | Ошибка | Массив пуст |  |
| **9** | arr = null | Ошибка | Создайте массив |  |

**10. Ввод целого числа с клавиатуры**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Номер теста** | **Входные данные** | **Ожидаемый результат** | **Выходные данные** | **Примечание** |
| **1** | 5 | 5 | 5 |  |
| **2** | -5 | -5 | -5 |  |
| **3** | 0 | 0 | 0 |  |
| **4** | фывфыв | Ошибка | Неверное число |  |
| **5** | 1,2345 | Ошибка | Неверное число |  |
| **6** | -5,1 | Ошибка | Неверное число |  |
| **7** | 22222222222222 | Ошибка | Неверное число |  |

**11. Создание массива чётных элемента из исходного массива**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Номер теста** | **Входные данные** | **Ожидаемый результат** | **Выходные данные** | **Примечание** |
| **1** | 5 | Пустой массив | Пустой массив |  |
| **2** | -2 3 5 6 | -2 6 | -2 6 |  |
| **3** | 0 | 0 | 0 |  |
| **4** | arr=[] | Пустой массив | Пустой массив |  |
| **5** | arr = null | Ошибка | Массив не инициализирован |  |

**12. Создание массива нечётных элемента из исходного массива**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Номер теста** | **Входные данные** | **Ожидаемый результат** | **Выходные данные** | **Примечание** |
| **1** | 5 | 5 | 5 |  |
| **2** | -2 3 5 6 | 3 5 | 3 5 |  |
| **3** | 0 | Пустой массив | Пустой массив |  |
| **4** | arr = [] | Пустой массив | Пустой массив |  |
| **5** | arr = null | Ошибка | Массив не инициализирован |  |