**НЕГОСУДАРСТВЕННОЕ ОБРАЗОВАТЕЛЬНОЕ ЧАСТНОЕ** **УЧРЕЖДЕНИЕ ВЫСШЕГО ОБРАЗОВАНИЯ** **«МОСКОВСКИЙ ФИНАНСОВО-ПРОМЫШЛЕННЫЙ УНИВЕРСИТЕТ** **“СИНЕРГИЯ”»**

|  |  |  |
| --- | --- | --- |
| **Факультет/Институт** |  | Синергия |
|  |  | (наименование факультета/ Института) |
| **Направление/специальность** |  | Информационные системы и программирование |
| **подготовки:** |  | (код и наименование направления /специальности подготовки) |
| **Форма обучения:** |  | очная |
|  |  | (очная, очно-заочная, заочная) |
|  |  |  |

**Отчет по лабораторной работе №3**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **на тему** |  | Рекурсивные функции | | |
|  |  | (наименование темы) | | |
|  |  |  | | |
| **по дисциплине** | | |  | Разработка программных модулей |
|  | | |  | (наименование дисциплины) |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Обучающийся** |  | Рахимов.А |  |  |
|  |  | (ФИО) |  | (подпись) |
| **Группа** |  | ДКИП – 205 прог |  |  |
|  |  |  |  |  |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Преподаватель** |  | Сиберев И.В. |  |  |
|  |  | (ФИО) |  | (подпись) |

**Москва 2025 г.**

**Лабораторная работа №3. «**Рекурсивные функции**»**

Цель работы: изучить способы реализации алгоритмов с использованием рекурсии.

1.3. Индивидуальные задания.

**2.** Удалить из созданного списка элементы с четными числами.

Код:

using System;

using System.Collections.Generic;

using System.Linq;

class Program

{

static void Main()

{

// Создаем список чисел

List<int> numbers = new List<int> { 1, 2, 3, 4, 5, 6, 7, 8, 9, 10 };

Console.WriteLine("Исходный список:");

Console.WriteLine(string.Join(", ", numbers));

// Удаляем четные числа (вариант 1 - с использованием RemoveAll)

numbers.RemoveAll(x => x % 2 == 0);

Console.WriteLine("\nСписок после удаления четных чисел (вариант 1):");

Console.WriteLine(string.Join(", ", numbers));

// Восстанавливаем исходный список для демонстрации второго варианта

numbers = new List<int> { 1, 2, 3, 4, 5, 6, 7, 8, 9, 10 };

// Удаляем четные числа (вариант 2 - с использованием LINQ Where)

numbers = numbers.Where(x => x % 2 != 0).ToList();

Console.WriteLine("\nСписок после удаления четных чисел (вариант 2):");

Console.WriteLine(string.Join(", ", numbers));

// Восстанавливаем исходный список для демонстрации третьего варианта

numbers = new List<int> { 1, 2, 3, 4, 5, 6, 7, 8, 9, 10 };

// Удаляем четные числа (вариант 3 - с использованием цикла)

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

{

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

{

numbers.RemoveAt(i);

}

}

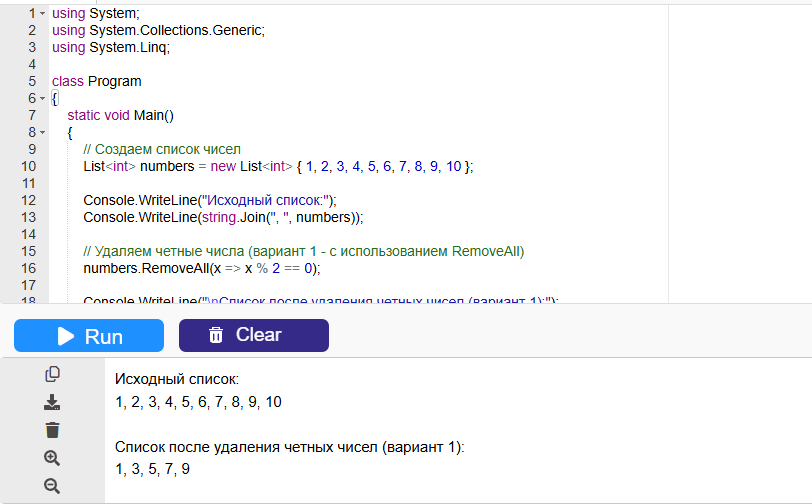
Console.WriteLine("\nСписок после удаления четных чисел (вариант 3):");

Console.WriteLine(string.Join(", ", numbers));

}

}

Код на компиляторе:



**3.4. Контрольные вопросы**

1. **Стек** — это структура данных, работающая по принципу **LIFO** ("последним вошёл, первым вышел").
   * Основные операции:
     + Push (добавить на вершину),
     + Pop (удалить с вершины),
     + Peek (посмотреть вершину).
2. **Рекурсивный тип данных** — это тип, который содержит ссылку на самого себя (например, узлы дерева или связного списка).
3. **Да, можно**, но стандартный стек (Stack<T>) не поддерживает прямое удаление элементов не с вершины.
   * **Решение:**
     + Временный стек для сохранения элементов,
     + Удаление нужного элемента,
     + Возврат остальных обратно.