Министерство науки и высшего образования Российской Федерации

Федеральное государственное бюджетное образовательное учреждение высшего образования

«Российский химико-технологический университет имени Д.И. Менделеева»

Кафедра информационных компьютерных технологий

ОТЧЕТ ПО ЛАБОРАТОРНОЙ РАБОТЕ № 3

Выполнила студентка группы КС-36 Битарова Эмма Олеговна

Ссылка на репозиторий: https://github.com/MUCTR-IKT-CPP/

Приняли: Пысин Максим Дмитриевич

Дата сдачи: 27.11.2022

Оглавление

[Описание задачи. 2](#_Toc63548272)

[Описание метода/модели. 2](#_Toc63548273)

[Выполнение задачи. 2](#_Toc63548274)

[Заключение. 2](#_Toc63548275)

# Описание задачи.

# В рамках лабораторной работы необходимо изучить и реализовать стек, в соответствии со своим вариантом, при этом, все структуры должны: Использовать шаблонный подход, обеспечивая работу контейнера с произвольными данными, Реализовывать свой итератор с реализацией операторов ++ и !=, Обеспечивать работу стандартных библиотек и конструкции for each если она есть в языке, Проверку на пустоту и подсчет количества элементов, Операцию сортировки с использованием стандартной библиотеки.

# Описание метода/модели.

# Стек сам по себе является частным случаем списка, поэтому часто его реализация является просто модификацией над уже существующими в библиотеке типом списка, которая просто запрещает доступ к определенным операциям, и добавляя новую.

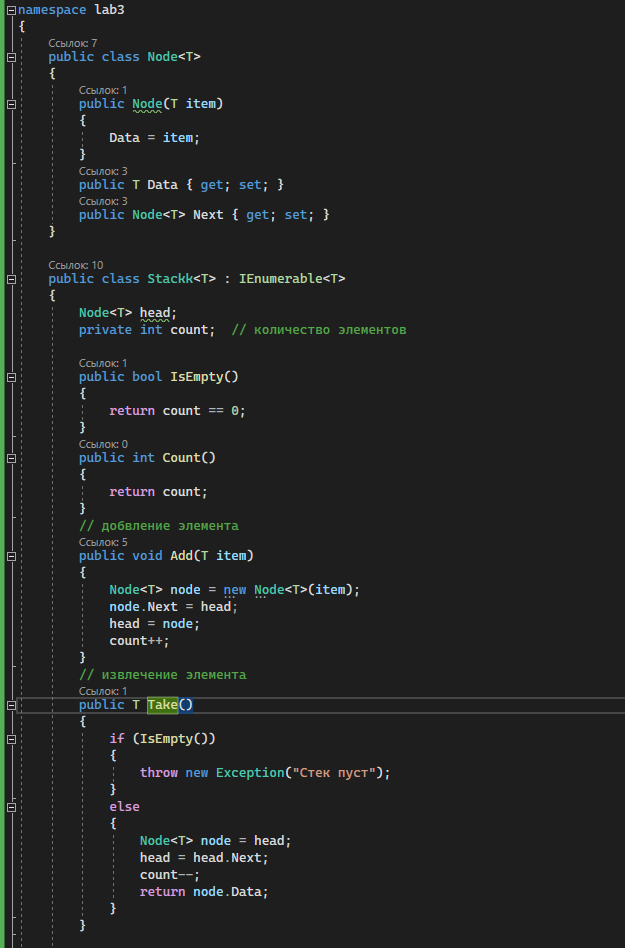
# Стек реализует идею звучащую следующим образом: “Последний пришел – первый ушел”

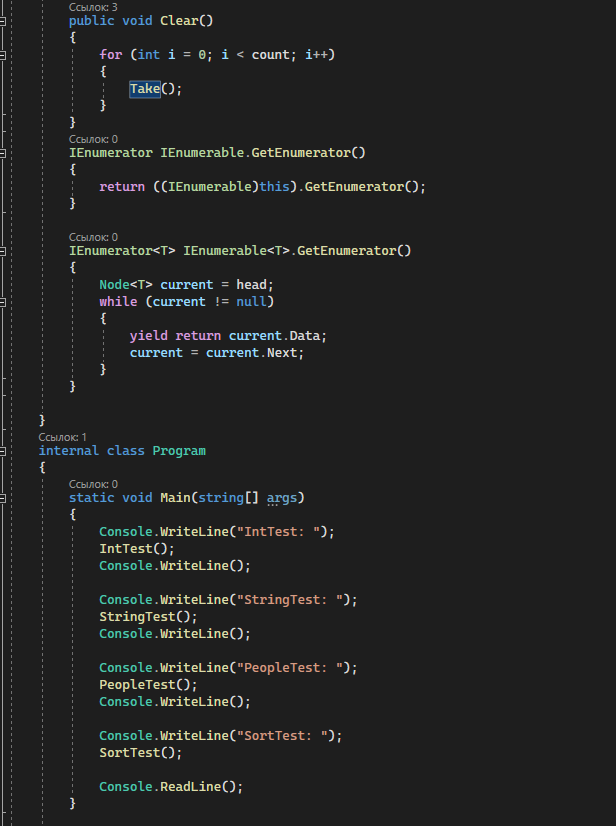
# Каждый элемент стека состоит из поля хранящего значение элемента стека, и поле указателя на следующий элемент стека. Это соответствует однонаправленному списку, однако в отличии от него, начало стека является его же и концом

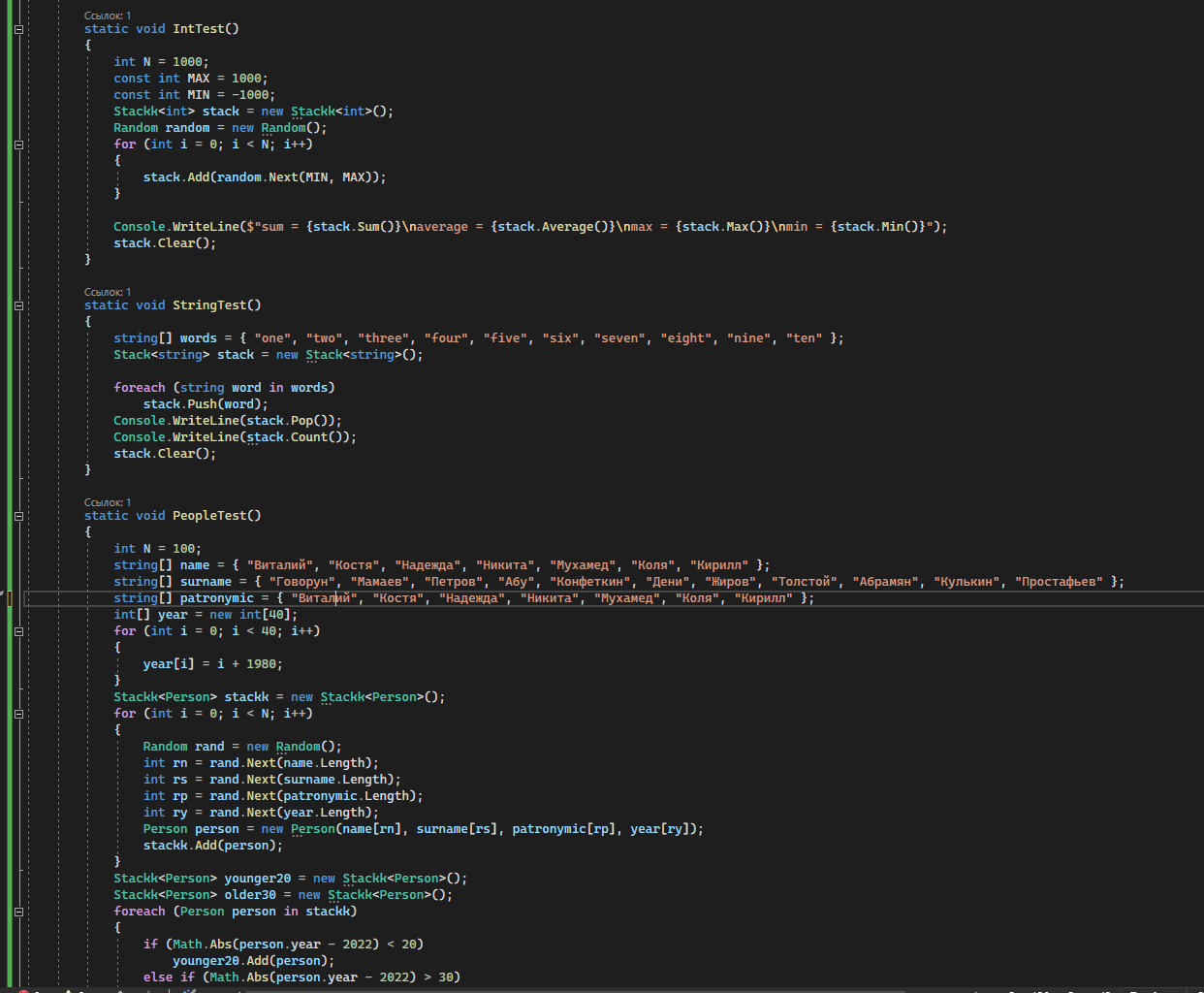
# Выполнение задачи.

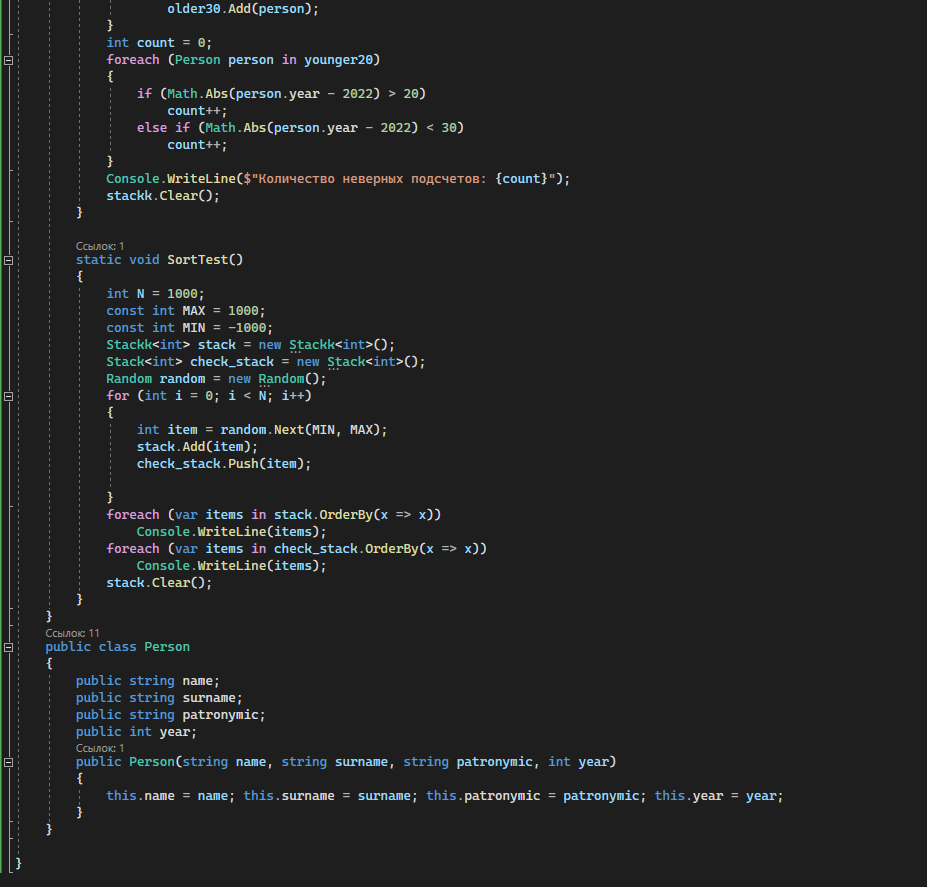
Язык: C#.

Были созданы следующие классы: Node - содержащий сам элемент; Stack со следующими методами: IsEmpty(проверка на пустоту), Count(подсчет элементов), Add(добавление в начало), Take(взятие элемента), Clear(очищение стека); Program со следующими функциями: IntTest(заполнение 1000 числами в диапазоне от -1000 до 1000, подсчет их суммы, среднего арифмитического, нахождение максимального и минимального значения), StringTest(проверка вставки), PeopleTest(заполнение контейнера), SortTest(заполнение и сортировка контейнера).









Изображение выглядит как текст

Автоматически созданное описание

Рисунок . Результат работы

# Заключение.

Стек позволяет управлять памятью эффективным образом, но если будет нужна частая работа с элементами стека из середины, то выбор стоит сделать в пользу другого контейнера.