МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ РОССИЙСКОЙ ФЕДЕРАЦИИ

ФЕДЕРАЛЬНОЕ государственное БЮДЖЕТНОЕ

образовательное учреждение

высшего образования

«НОВОСИБИРСКИЙ ГОСУДАРСТВЕННЫЙ ТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ»

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Кафедра защиты информации

*Изображение выглядит как текст, Шрифт, снимок экрана, логотип

Автоматически созданное описание*

**ЛАБОРАТОРНАЯ РАБОТА №2**

**«**Алгоритмы и абстрактные структуры данных**»**

**по дисциплине: «*Программирование*»**

|  |  |
| --- | --- |
| Выполнил:  Студент гр. «АБс-324», «АВТФ»  *Петров Максим Игоревич*  «31» октября 2024г  \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_  (подпись) | Проверил:  *Ассистент кафедры ЗИ*  *Исаев Глеб Андреевич*  «\_\_\_» \_\_\_\_\_\_ 2024г  \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_  (подпись) |

Новосибирск 2024

**Цели и задачи работы:** изучение алгоритмов работы с абстрактными структурами данных.

**Задание к работе:** самостоятельно решить задачи в соответствии с индивидуальным вариантом.

Задания:

1. Стек. Вариант 3. Преобразование инфиксной нотации в постфиксную.
2. Реализовать основные операции со множеством: добавление элемента, удаление элемента, проверка наличия элемента в множестве.
3. Множество. Вариант 3. Дополнительные операции Множества.
4. Массив. Вариант 5. Контрольная по ударениям.
5. Двоичное дерево. Вариант 1. Удаление узла.
6. Хеш-таблица. Вариант 3. Самая длинная подстрока.

Контрольные примеры

Стек, вариант 3:

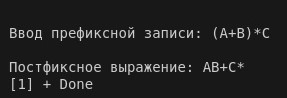
Вводим инфиксные выражения: A+B\*C, (A+B)\*C.

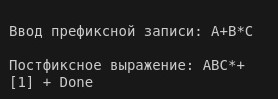


Изображение выглядит как текст, Шрифт, черный, снимок экрана

Автоматически созданное описание

Вывод программы постфиксного выражения:

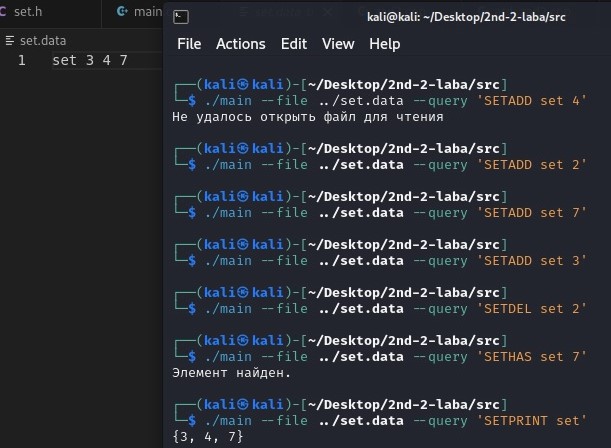




Множество:

Открываем терминал в директории где находится наш проект и используем команду: ./названиеФайла –file названиеФайла –query “Команда”.

Добавляем элементы 4, 2, 7, 3. Далее удаляем элемент 2 и проверяем наличие элемента 7 в множестве, и выводим множество.

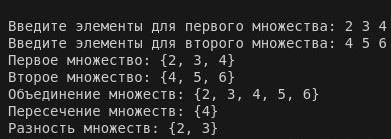


Множество, вариант 3:

Вводим два множества.



Вывод программы, которая показывает дополнительные операции со множеством:



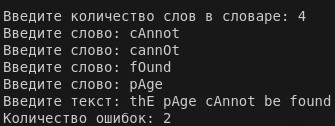
Массив, вариант 5:

Вводим слова и выделяем их ударение.

Изображение выглядит как текст, Шрифт, снимок экрана, черный

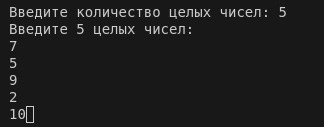
Автоматически созданное описание

Вывод программы, которая показывает количество ошибок, допущенных в словах:

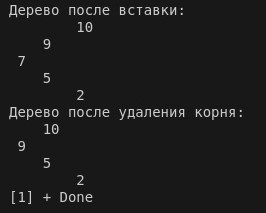


Двоичное дерево, вариант 1:

Вводим количество целых чисел, которые будут в дереве. Далее вводим сами числа.



Вывод программы, после удаления вершины и дерево остается корректным:

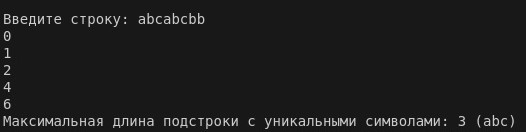


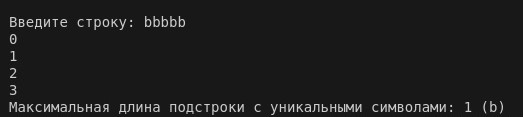
Хеш-таблица, вариант 3:

Вводим строку, в которой все символы различны.

Вывод программы, которая определяет самую длинную подстроку:





**Вывод**

В ходе выполнения работы я использовал свои реализованные структуры данных и изучил алгоритм работы с абстрактными структурами данных. И реализовал собственное множество.

Ссылка на гит: <https://github.com/Mixassss/2nd-2-laba.git>