Федеральное государственное бюджетное образовательное учреждение

высшего образования

«Алтайский государственный технический университет

им. И.И. Ползунова»

Факультет информационных технологий

Кафедра информатики, вычислительной техники

и информационной безопасности

Направление Информатика, вычислительная техника

Отчет защищен с оценкой\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Преподаватель Н.Д. Бубнова (подпись)

“\_\_\_\_”\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_2023 г.

(дата)

Отчёт

по лабораторной работе № 2

Связные списки

По дисциплине Структуры данных

ЛР 09.03.01 7.00 О

Студент группы ИВТ-21 И.А. Мелков

Преподаватель старший преподаватель Н.Д. Бубнова

Барнаул 2023

**Лабораторная работа № 2**

**Связные списки**

Задание

1.Выбрать вариант. Номер варианта должен совпадать с номером в списке группы в личном кабинете

2.Ознакомиться с постановкой задачи

3.Разработать алгоритм решения задачи.

4.Написать программу для ее реализации

5.Разработать тесты

6.Исходные данные поместить в файл input.dat

7.Результаты вывести на экран. Исходные данные и результаты вывести также в выходной файл output.dat

8.Оформить отчет

Задание принял к исполнению:

Студент  И.А. Мелков

(подпись)

**Вариант №16**

**Задание №1.**

**Заданы два списка. Пополнить первый уникальными элементами второго.**

//описание код в выводе.

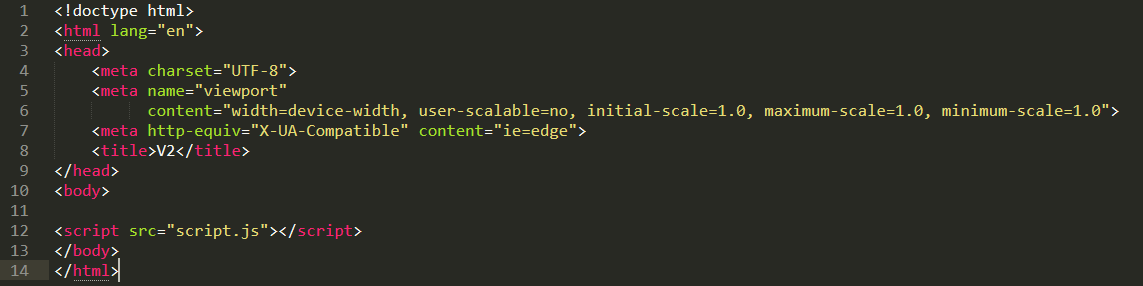


Рисунок 1 – код программы html (структура файла)

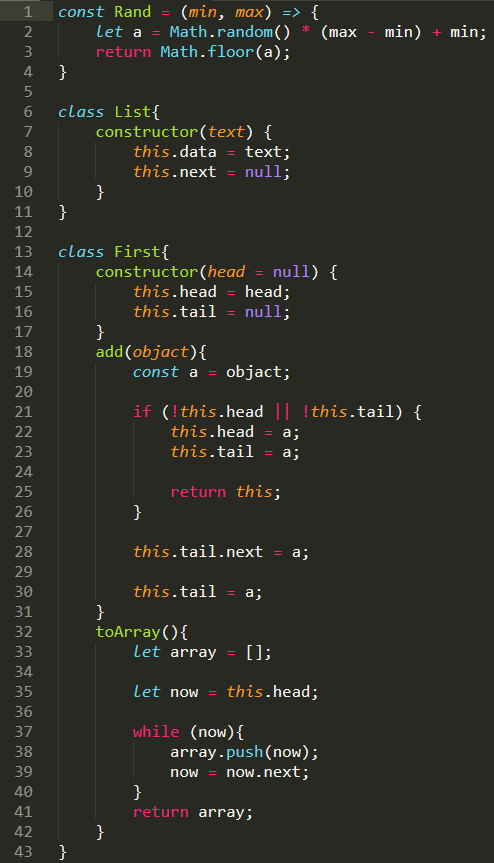


Рисунок 2 – код программы js (Вариант 1)

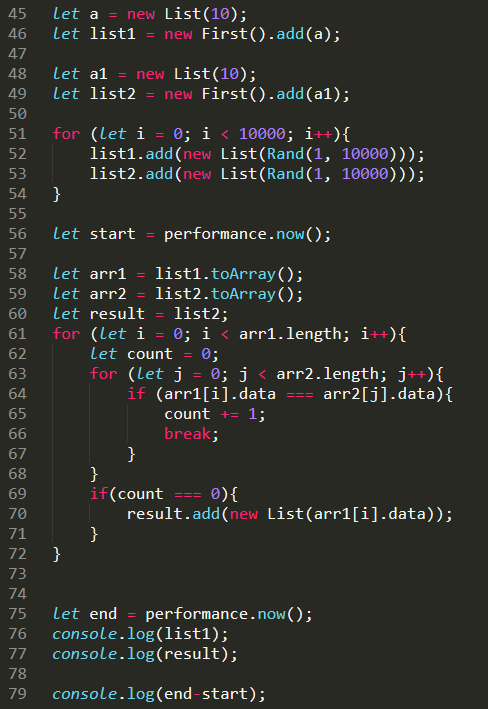


Рисунок 3 – код программы js (Вариант 1)

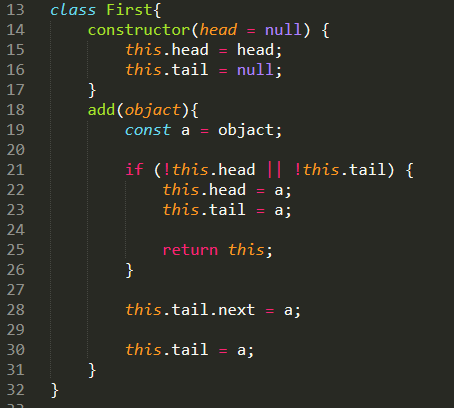


Рисунок 4 – код программы js (Вариант 2)

Во 2 варианте программы я убрал метод перевода списка в массив из класса списка.

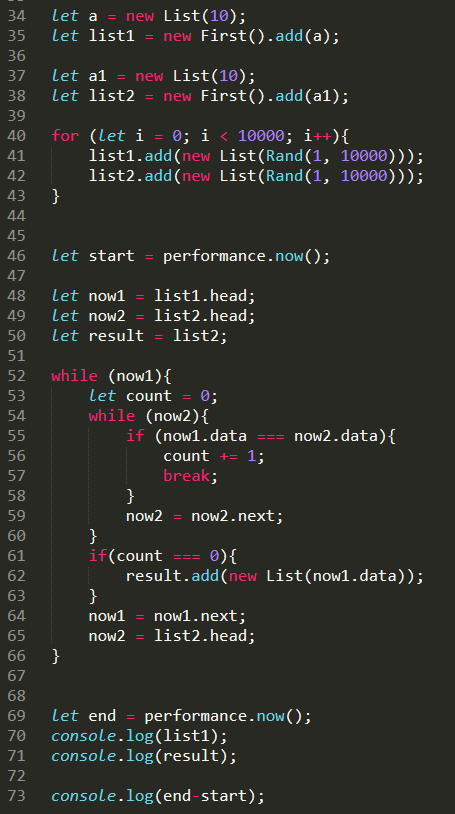


Рисунок 5 – код программы js (Вариант 2)

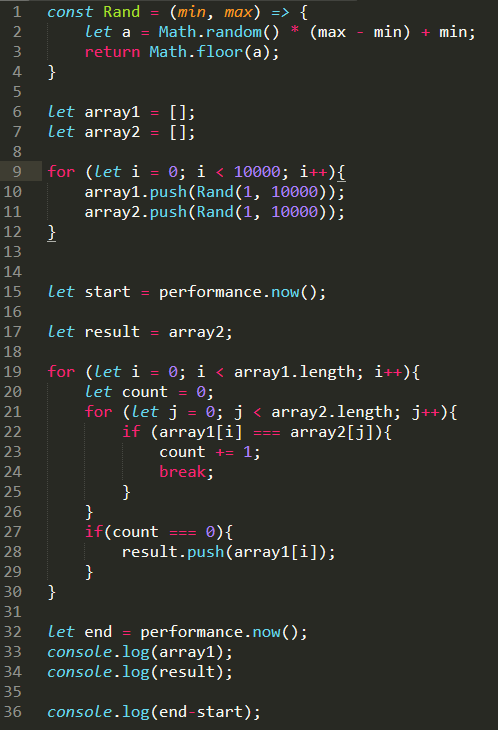


Рисунок 6 – код программы js (Вариант 3)

**Тесты**

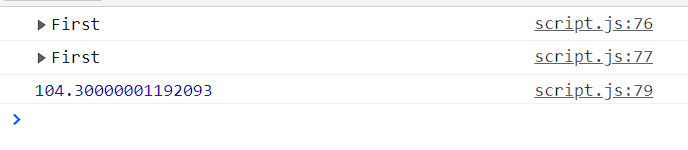


Рисунок 7 – Замеры по времени (Вариант 1)



Рисунок 8 – Замеры по времени (Вариант 2)



Рисунок 9 – Замеры по времени (Вариант 3)

Вывод: в данной лабораторной работе я столкнулся с тем, что связный список работает сильно медленней, чем работать с массивами. Я написал 3 варианта работы для сравнения скорости. В 1 варианте я создал связный список + метод который переводит его в массив, для проверки всё переводилось в массив и в конечном итоге от связных списков я уходил и это решение является супер неэффективным. В 2 варианте поиск уникальных элементов я реализовал через цикл while и к массивам не прибегал. В 3 варианте, для интереса, я просто работал с массивом. Результат меня удивил. Правильная работа со связными списками занимает в 5-6 раз больше времени чем с массивами, даже при условии, что мы учитываем время перевода списка в массив. В сравнении с обычными массивами, не прибегая к связным спискам, алгоритм выполняется в 7 раз медленней. Потом я понял, что это нормально для данной структуры данных, так как она не рассчитана на быстродействие, а рассчитана на экономию памяти. Допустим в Веб-разработке, связные списки почти не используются. Но это востребовано в разработке ПО для микроконтроллеров, работы с биг-дата и т.д..