Санкт-Петербургский Национальный Исследовательский Университет Информационных Технологий, Механики и Оптики

Кафедра Систем Управления и Информатики

**Лабораторная работа №1**

**Вариант №1**

Выполнили:

Новиков Н. В.

Ходакова М. А.

Борисевич А. В.

Проверил: Мусаев А. А.

Санкт-Петербург,

2023

**Содержание**

[ВВЕДЕНИЕ 3](#_Toc146569853)

[1. Программа бинарного поиска 4](#_Toc146569854)

[2. Программа для определения одногруппников 4](#_Toc146569855)

[2.2 Описание программы 4](#_Toc146569856)

[2.2 Работа алгоритма 5](#_Toc146569857)

[3. Граф по второму заданию 6](#_Toc146569858)

[СПИСОК ИСТОЧНИКОВ 7](#_Toc146569859)

[ЗАКЛЮЧЕНИЕ 8](#_Toc146569860)

# ВВЕДЕНИЕ

Целью лабораторной работы №1 было знакомство с графами. Были поставлены следующие задачи:

1. Написание программы бинарного поиска
2. Создание программы для определения одногруппников
3. Составить граф на основе программы из 2 задачи

# 1. Программа бинарного поиска

Создадим некоторый список и зададим значение переменной x из списка. Тогда переменная k – счётчик количества шагов для нахождения числа бинарным поиском. Обозначим переменными left и right левую и **правую границы поиска соответственно. Используем цикл while для** нахождения числа, который работает до тех пор, пока левая граница не больше правой. При входе в цикл счетчик количества шагов увеличивается на 1, т. к. шаг будет выполнен. Определяем середину области поиска (переменная mid) как среднее арифметическое границ. Если элемент списка с индексом mid является искомым, то выходим из цикла. В противном случае, если элемент списка с индексом mid меньше искомого числа, то передвигаем левую границу поиска, присваивая ей значение mid+1, т. к. искомое число находится правее середины списка и элемент mid не является искомым. Иначе элемент находится левее элемента списка с индексом mid, значит, необходимо сдвинуть правую границу, присвоить переменной right значение mid-1. Таким образом, выход из цикла будет совершен, когда будет найдена переменная или когда цикл пройдется по всем элементам массива. В конце будет выведено количество шагов.

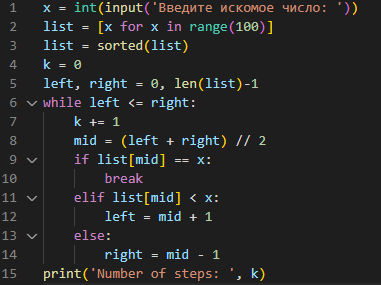


Рисунок 1 - Бинарный поиск

# 2. Программа для определения одногруппников

## 2.2 Описание программы

Одной из целей лабораторной работы было создание программы, которая позволяет по определенным вопросам найти одногруппника. Программа была написана с помощью python без использования библиотек. Для хранения всех одногруппников хватило 7 вопросов.

## 2.2 Работа алгоритма

Сначала создаем множество, в которое будут помещаться студенты ответивших положительно на тот или иной вопрос. После создаем множество всех студентов. Алгоритм проходится по всех студентам и добавляет его во множество всех возможных. Если студент ответил на i-ый вопрос положительно, то он добавляется во множество i из изначального массива (Рисунок 2). После создания списка студентов программа проходиться по всем вопросам в файле и ожидает ввода ответа от пользователя. Если студент отвечает положительно, то происходит сужение круга студентов. Пересекается множество всех студентов со множеством студентов, ответивших на вопрос положительно. Искомый студент входит в оба множества. В случае отрицательного ответа, то из множества возможных студентов вычитается множество студентов ответивших положительно. Это подразумевает, что первый студент входит в первое множество, но не входит во второе (Рисунок 3). Если же длина всех возможных студентов окажется равна 1, то значит, что программа нашла загаданного студента. Если же длина равна 0, то это означает, что такого студента нет (Рисунок 4).

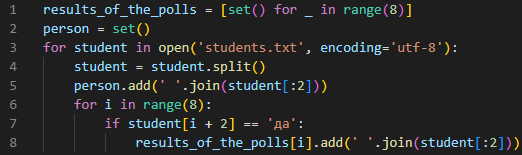


Рисунок 2 - Перебор всех студентов

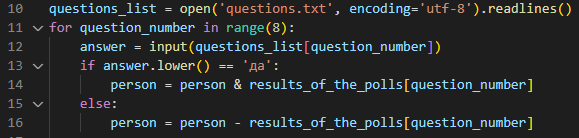


Рисунок 3 - работа со множеством студентов

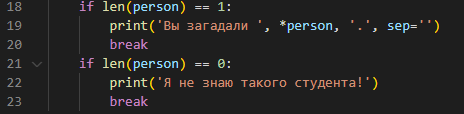


Рисунок 4 - Возможные исходы программы

# 3. Граф по второму заданию

Последним заданием лабораторной работы было создание графа по второму заданию. Этот граф – граф-дерево (Рисунок 5), так как любые 2 его вершины соединены лишь одним маршрутом. Каждый поворот влево, в котором означает ответ “нет”, а в право “да”.

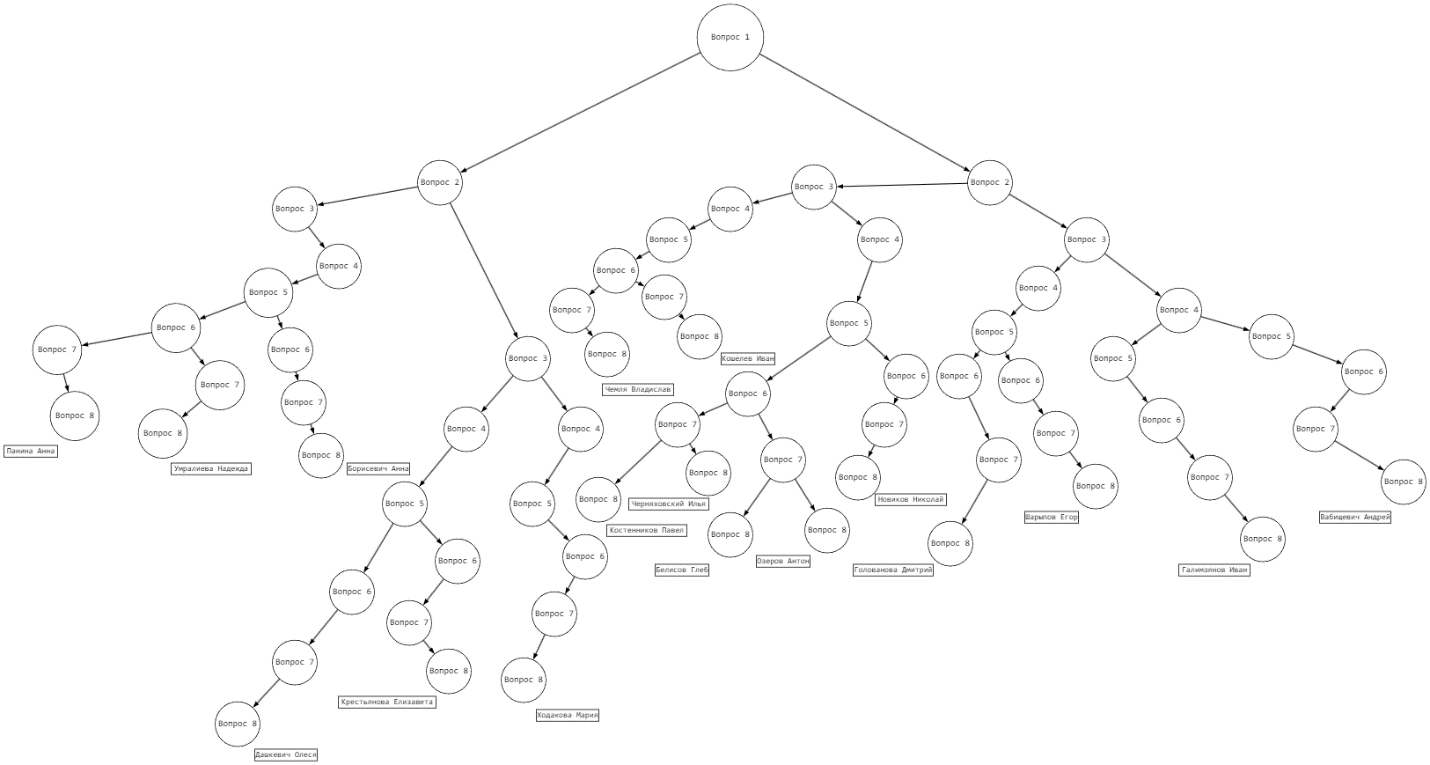


Рисунок 5 - Граф одногруппников

# СПИСОК ИСТОЧНИКОВ

1. Github репозиторий лабораторной работы. [Электронный ресурс] <https://github.com/sedm1/lab-1-aisd>

# ЗАКЛЮЧЕНИЕ

В ходе выполнения лабораторной работы №1 были получены навыки работы с графами, а также с алгоритмом бинарного поиска и работы со множеством.