ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ АВТОНОМНОЕ ОБРАЗОВАТЕЛЬНОЕ УЧРЕЖДЕНИЕ ВЫСШЕГО ОБРАЗОВАНИЯ

«НАЦИОНАЛЬНЫЙ ИССЛЕДОВАТЕЛЬСКИЙ УНИВЕРСИТЕТ ИТМО»

Отчёт по лабораторной работе № 4

«Задача о покрытии множеств»

Выполнил работу

Урядов Валерий

Академическая группа №J3112

Принято

Должность, звание Фамилия Имя преподавателя

Санкт-Петербург

2024

1. **Введение**

Цель работы: Реализация алгоритма для поиска минимального покрытия множества.

1. **Задачи**

Изучить принцип задачи минимального покрытия множества.

Реализовать алгоритм полного перебора для нахождения минимального подмножества.

Выполнить тестирование программы на различных наборах данных.

1. **Теоретическая подготовка**

В этом разделе вам необходимо изложить все теоретические аспекты, которые необходимы для выполнения работы, в том числе используемые типы данных, алгоритмы и пр.

1. **Реализация**

Код программы реализует алгоритм, который перебирает все возможные комбинации подмножеств для нахождения минимального покрытия. Основные элементы реализации:

Функция covers проверяет, покрывает ли текущее объединение подмножеств все элементы множества AAA.

Функция set\_cover выполняет полный перебор всех комбинаций с использованием битовой маски.

Проверка времени работы программы, выбрасывающая исключение, если работа алгоритма превышает один час.

1. **Экспериментальная часть**

Программа была протестирована на нескольких наборах данных. Рассмотрим один из примеров:

*Входные данные:*

Множество A={1,2,3}A = \{1, 2, 3\}A={1,2,3}

Коллекция подмножеств B={{1,2},{2,3},{3,4}}B = \{\{1, 2\}, \{2, 3\}, \{3, 4\}\}B={{1,2},{2,3},{3,4}}

*Результат:*

Минимальное подмножество, покрывающее множество AAA: {{1,2},{2,3}}\{\{1, 2\}, \{2, 3\}\}{{1,2},{2,3}}

Теоретическая сложность алгоритма O(2N)O(2^N)O(2N), где NNN — количество подмножеств.

1. **Заключение**

В ходе выполнения работы реализован алгоритм полного перебора для поиска минимального покрытия множества. Программа успешно проходит тестирование на различных примерах. В дальнейшем можно оптимизировать алгоритм с помощью жадных методов или методов динамического программирования для улучшения производительности при больших объёмах данных.

1. **Приложения**

