ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ АВТОНОМНОЕ ОБРАЗОВАТЕЛЬНОЕ УЧРЕЖДЕНИЕ ВЫСШЕГО ОБРАЗОВАНИЯ

«НАЦИОНАЛЬНЫЙ ИССЛЕДОВАТЕЛЬСКИЙ УНИВЕРСИТЕТ ИТМО»

Отчёт по лабораторной работе № 7

«Жадные алгоритмы»

Выполнил работу

Афанасьев Николай

Академическая группа №J3113

Принято

Должность, звание Фамилия Имя преподавателя

Санкт-Петербург

2024

**Структура отчёта:**

* Введение

Целью данной лабораторной работы является изучение принципов работы жадных алгоритмов, понимание их преимуществ и ограничений, а также определение областей, где их применение наиболее эффективно.

Задачи, решаемые в рамках данной работы:

1. Изучение теоретических основ жадных алгоритмов: определение, основные характеристики, принцип работы.
2. Реализация на практике одного или нескольких примеров жадных алгоритмов для решения задачи на платформе leetcode.
3. Подсчет сложности и затрат памяти полученного алгоритма.

* Теоретическая подготовка

1. Основные понятия:
2. Жадный выбор: Принцип принятия локально оптимального решения на каждом шаге с надеждой на получение глобально оптимального решения.
3. Оптимальная подструктура: Свойство задачи, при котором оптимальное решение задачи содержит в себе оптимальные решения подзадач. Наличие оптимальной подструктуры является необходимым, но недостаточным условием для применения жадного алгоритма.
4. Используемые типы данных и структуры:
   1. Вектор (std::vector): Для хранения последовательности бросков кубика rolls. В данном случае std::vector<int> используется для хранения целых чисел.
   2. Неупорядоченное множество (std::unordered\_set): Ключевая структура данных в данной реализации. std::unordered\_set<int> используется для хранения уникальных значений бросков кубика, встретившихся в текущей подпоследовательности. Важно отметить, что unordered\_set обеспечивает быстрый (в среднем O(1)) поиск, вставку и удаление элементов.
5. Алгоритмы и методы:
   1. Жадный алгоритм: Основная идея алгоритма заключается в том, чтобы на каждом шаге проверять, встретились ли все возможные значения бросков кубика от 1 до k в текущей подпоследовательности. Если все значения встретились, это означает, что мы можем построить любую подпоследовательность длиной 1. Следовательно, минимальная невозможная подпоследовательность будет иметь длину на 1 больше.

* Реализация

Описание алгоритма: Алгоритм использует unordered\_set для отслеживания уникальных бросков кубика. При проходе по массиву rolls каждое значение добавляется в множество. Если размер множества достигает k, это означает, что мы встретили все возможные значения бросков, и, следовательно, можем сформировать любую последовательность длины 1. В этом случае счетчик ans (длина минимальной невозможной последовательности) увеличивается на 1, а множество очищается.

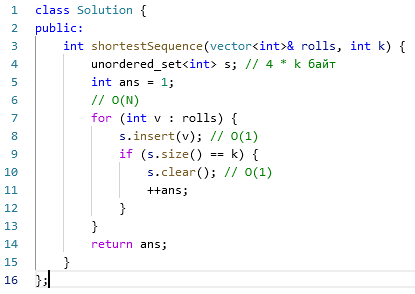


Рис.1

* Экспериментальная часть

Подсчёт по памяти (только для циклов и сложных структур) – Алгоритм использует unordered\_set, который в худшем случае может хранить до k элементов. Следовательно, пространственная сложность составляет O(k). см рис.1

Подсчёт асимптотики (только для циклов и сложных структур) –   
Алгоритм имеет временную сложность O(n), где n — длина вектора rolls. Это обусловлено однократным проходом по входным данным. Операции с unordered\_set (вставка, поиск, очистка) в среднем имеют сложность O(1). см рис.1

* Заключение

В ходе данной работы был рассмотрен жадный алгоритм для решения задачи "Shortest Impossible Sequence of Rolls". Было показано, что использование подходящих структур данных, таких как unordered\_set, позволяет эффективно реализовать жадные стратегии и достичь оптимальной производительности.

* Приложения

Изображение выглядит как текст, программное обеспечение, Шрифт, число

Автоматически созданное описание

<https://habr.com/ru/articles/120343/>  
 <https://education.yandex.ru/handbook/algorithms/article/zhadnye-algoritmy>