

Университет ИТМО

Факультет программной инженерии и компьютерной техники

Лабораторная работа №4
по «Алгоритмам и структурам данных»
Базовые задачи

Выполнил:

Студент группы Р3230

Асташин С.С.

Преподаватели:

Косяков М.С.

Тараканов Д.С.

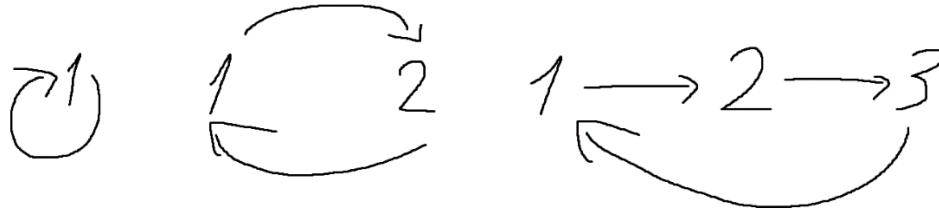
Санкт-Петербург

2022

Задача N «Свинки-копилки»

Пояснение к примененному алгоритму:

Представим наших свинок в виде орграфа. Несложно догадаться, что разбивать какую-то копилку придётся только в том случае, если у нас цикл (любой из следующих вариантов):



Будем хранить зависимость свинок друг от друга в `map<int,int> all_pigs`. Потом просто пойдём по НЕПРОВЕРЕННЫМ свиньям циклом от 1 до `n`, и для каждой свиньи во вложенном цикле будем пытаться пройти дальше по стрелкам, при этом сохраняя, через каких свинок прошли. Если встретится повторившийся индекс, то увеличим счётчик разбитых свинок, а всех свинок, которые входят в цикл, отметим как проверенных.

Оценка сложности: $O(n^2)$.

Задача O «Долой списывание!»

Пояснение к примененному алгоритму:

Это задача на раскраску графа в 2 цвета (допустим, красный и синий). Каждая вершина в `vector<Vertex> graph` хранит свой цвет (`none`, `red`, или `blue`), список соседей и `bool neighbours_colored`. Алгоритм проверки – фактически, поиск в глубину. Мы раскрашиваем нулевую вершину в красный, а потом рекурсивно вызываем функцию `bool draw(int i)` на тех вершинах, у которых есть нераскрашенные соседи.

Оценка сложности: $O(n+m)$.

Задача Р «Авиаперелёты»

Пояснение к примененному алгоритму:

Сначала заметим два факта: делать размер бака меньше минимального расхода топлива между городами бессмысленно, поскольку мы тогда никуда не долетим, а делать размер больше максимального – тоже, потому что у нас есть возможность дозаправки, а лишний вес самолёту не особо нужен. Найдём эти два числа: минимальное и максимальное количество топлива, которое надо затратить на перелёт.

Далее запускаем бинарный поиск в найденном отрезке. Внутри бинарного поиска будем пытаться облететь все города при текущем размере бака: для этого дважды вызовем поиск в глубину на нашем графе (дважды, поскольку значения расхода по горизонтали и по вертикали в таблице не всегда совпадают). Два вызова поиска в глубину должны нам вернуть два числа – количество городов, которые удалось облететь. Если оно равно $2N$, то это значит, что топлива нам хватило, и можно попытаться найти меньший объём бака ($r = mid$), иначе – больший объём бака ($l = mid + 1$).

Оценка сложности: $O(\log(\max - \min)) * (O(n^2) + O(n^2)) = O(n^2 * \log(\max - \min))$.