XXIII Командная олимпиада школьников Санкт-Петербурга по программированию

25 октября 2015 года

Задача А. «Атомы»

Задача А. «Атомы»

- Идея задачи — Дмитрий Филиппов
- Подготовка тестов — Николай Будин
- Разбор задачи — Николай Будин

Постановка задачи

- Есть последовательность чисел.
- Поступают запросы двух типов.
- Прибавить число на отрезке.
- Найти максимальную длину арифметической прогрессии с $d = 1$ на отрезке.

Решение задачи

- Будем рассматривать последовательность разностей соседних чисел.
- Теперь нужно искать максимальную длину последовательности из единиц на отрезке.
- Запрос на прибавление на отрезке привращается в два запроса на прибавление к элементу.
- Реализуем с помощью дерева отрезков.

Информация, хранимая в вершине

- Максимальная длина последовательности единиц на отрезке.
- Количество единиц с левого края на отрезке.
- Количество единиц с правого края на отрезке.

Пересчет информации

- $Max = \max(Max_l, Max_r, R_l + L_r)$.
- $L = L_l$, если $L_l < Length_l$.
- $L = Length_l + L_r$, если $L_l = Length_l$.
- $R = R_r$, если $R_r < Length_r$.
- $R = Length_r + R_l$, если $R_r = Length_r$.

Обновление информации в элементе

- Если значение элемента равно 1, $Max = 1$, $L = 1$ и $R = 1$.
- Иначе, $Max = 0$, $L = 0$ и $R = 0$.

Задача B. «День рождения»

Задача B. «День рождения»

- Идея задачи — Михаил Дворкин
- Подготовка тестов — Михаил Дворкин
- Разбор задачи — Михаил Дворкин

Постановка задачи

- Загадана дата, Аня знает число, а Боря — месяц.
- Опубликован список дат, включающий загаданную.
- Аня: «Я не знаю дату, но и Боря не знает.»
- Боря: «Я раньше не знал, но теперь знаю.»
- Аня: «Теперь я тоже знаю.»
- Найти загаданную дату.

Решение задачи

- Будем моделировать размышления Ани и Бори во всех возможных ситуациях.
- Сначала все даты из опубликованного списка пометим как *допустимые*.
- Постепенно даты будут помечаться как недопустимые, и останется одна дата — искомая.

Решение задачи: фаза 1 — фраза 1

- Аня: «Я не знаю дату, но и Боря не знает.»
- Рассмотрим каждую допустимую дату (d, m). Она остается допустимой, только если:
- с днем d есть хотя бы две допустимые даты;
- все допустимые даты с днём d таковы, что в их месяце есть хотя бы две допустимые даты.

Решение задачи: фаза 2 — фраза 2

- Боря: «Я раньше не знал, но теперь знаю.»
- Рассмотрим каждую допустимую дату (d, m). Она остается допустимой, только если:
- в месяце m есть ровно одна допустимая дата.

Решение задачи: фаза 3 — фраза 3

- Аня: «Теперь я тоже знаю.»
- Рассмотрим каждую допустимую дату (d, m). Она остается допустимой, только если:
- с днем d есть ровно одна допустимая дата.

- Поскольку входные данные корректны, после этого ровно одна дата останется допустимой.

Задача C. «Шушпанчики и кинотеатр»

Задача C. «Шушпанчики и кинотеатр»

- Идея задачи — Евгений Замятин
- Подготовка тестов — Илья Збань
- Разбор задачи — Илья Збань

Постановка задачи

- Есть матрица, некоторые клетки заняты, есть выделенная позиция (x, y).
- Нужно выбрать k соседних точек в одном ряду так, чтобы суммарное манхэттенское расстояние до выделенной было минимально.

Решение задачи

- Нас интересуют лишь ряды, на которых есть занятые места или выделенное место, и соседние с ними.
- Для одного ряда мы знаем занятые места в нем.
- Если между двумя соседними занятыми местами хотя бы k свободных, попробуем обновить ответ.

Решение задачи

- Если место y не в рассматриваемом отрезке, то нужно выбрать k ближайших с конца, ближайшего к y.
- Если место y в рассматриваемом отрезке, то нужно выбрать его, и жадно ближайших его соседей.
- Ответ считается как сумма одной или двух арифметических прогрессий.

Задача D. «Доставка футболок»

Задача D. «Доставка футболок»

- Идея задачи — Андрей Лопатин
- Подготовка тестов — Демид Кучеренко
- Разбор задачи — Демид Кучеренко

Постановка задачи

- Есть курьер, который посещает адресатов посылок строго по очереди.
- По приезду на место курьер ждет адресата не более k минут.
- Если в течении этих k минут дверь открывается, то курьер доставляет посылку в течении t минут с момента открытия двери.
- Определить время, через которое курьер доставит все посылки.

Решение задачи

- Промоделируем процесс. Будем подсчитывать текущее время и переходить от клиента к клиенту по очереди.
- Пусть мы приехали к очередному клиенту в момент времени a.
- Если клиент готов принять посылку во время меньше либо равное, чем $a + k$, то посылка будет достанена.

Решение задачи

- Если посылка не будет доставлена, то прибавляем ко времени ожидания k и переезжаем к следующему клиенту (добавляем соответствующее z_i).
- Если посылка будет доставлена, то определяем время, в которое клиент будет готов принять заказ.
- Оно равно максимуму из времени прибытия к клиенту a и времени готовности клиента s_i.
- Добавляем к максимуму время доставки t и отправляемся далее по маршруту.

Задача E. «Деление»

Задача E. «Деление»

- Идея задачи — Захар Войт
- Подготовка тестов — Николай Будин
- Разбор задачи — Николай Будин

Постановка задачи

- Есть два числа: n и m.
- Изменить минимальное количество цифр в первом, чтобы оно стало делиться на второе, или сказать, что это невозможно.

Решение задачи

- Разобьем задачу на две подзадачи.
- Первая — $m \leqslant \sqrt{n}$.
- Вторая — $\sqrt{n} < m$.

Решение первой подзадачи ($m \leqslant \sqrt{n}$)

- Подзадача решается с использованием метода динамического программирования.
- Состояние d_{ij}, где i — длина суффикса числа n, а j — остаток от деления на число m.
- В состоянии хранится минимальное количество цифр, которые нужно изменить на суффиксе длины i, чтобы этот суффикс стал давать остаток j при делении на m.

Решение первой подзадачи ($m \leqslant \sqrt{n}$)

- Для пересчета d_{ij} нужно рассмотреть состояния, получаемые из данного уменьшением длины суффикса на 1 и подстановкой всех возможных цифр в n на i-е с конца место.
- $j_{new} = (j - d \cdot 10^{i-1}) \mod m$, где d — значение цифры, подставленной в n.
- Причем, если подставленная цифра не совпадает с цифрой, стоящей этой позиции в числе n, для пересчета к этому значению нужно прибавить 1.

Решение первой подзадачи ($m \leqslant \sqrt{n}$)

- Так как результат не может иметь ведущих нулей, первая его цифра не должна равняться 0.
- Поэтому при пересчете суффикса длины n, не нужно подставлять цифру 0.

Решение первой подзадачи $(m \leqslant \sqrt{n})$

- Ответ — $d_{n,0}$
- Асимптотика — $O(m \cdot L_n \cdot N_d) = O(\sqrt{n} \cdot L_n \cdot N_d)$, где L_n — длина n, N_d — количество цифр, то есть 10.

Решение второй подзадачи ($\sqrt{n} < m$)

- Переберем k — частное ответа и m.
- Так как длина ответа равна длине n, длина частного не больше разности длин n и m.
- Разность длин n и m не больше $\left\lceil \frac{L_n}{2} \right\rceil$.
- Поэтому можно перебирать k, пока длина $m \cdot k$ не превышает длину n.
- Асимптотика — $O(10^{\lceil L_n \cdot 0.5 \rceil} \cdot L_n) = O(\sqrt{n} \cdot L_n)$, где L_n — длина n.

Решение задачи

- Крайний случай — ответ 0, нужно рассмотреть отдельно.

Задача F. «Милый общий делитель»

Задача F. «Милый общий делитель»

- Идея задачи — Демид Кучеренко, Евгений Замятин
- Подготовка тестов — Евгений Замятин
- Разбор задачи — Евгений Замятин

Постановка задачи

Найти общий делитель чисел a и b, сумма цифр которого максимальна.

Решение задачи

- Если $x \cdot y = n$ и $x \leqslant y$, то $x \leqslant \sqrt{n}$.
- Различных делителей у числа n не больше, чем $2 \cdot \sqrt{n}$.
- Можем перебрать все делители числа a.
- Среди всех делителей числа a выберем такой делитель d, что $b \vdots d$ и сумма цифр d максимальна.

Задача G. «Робот»

Задача G. «Робот»

- Идея задачи — Дмитрий Филиппов
- Подготовка тестов — Евгений Замятин
- Разбор задачи — Евгений Замятин

Постановка задачи

Изменить знак не более чем у k чисел из массива a_i, чтобы $|\sum a_i|$ был максимален.

Решение задачи

- Максимум $|\sum a_i|$ достигается, когда либо $\sum a_i$ максимальна, либо $\sum a_i$ минимальна.
- Попробуем максимизировать и минимизировать $\sum a_i$. После выберем лучший ответ.
- Чтобы максимизировать сумму, необходимо выбрать k максимальных по модулю отрицательных чисел и изменить их знак.
- Чтобы минимизировать сумму, необходимо выбрать k максимальных положительных чисел и изменить их знак.

Задача H. «Эстафета»

Задача H. «Эстафета»

- Идея задачи — Артем Васильев
- Подготовка тестов — Илья Збань
- Разбор задачи — Илья Збань

Постановка задачи

- Есть граф из $n+1$ вершины, нужно выбрать k циклических путей заданной длины, проходящих через 0.
- Нужно минимизировать суммарную длину циклов.

Решение задачи

- Будем решать задачу методом динамического программирования.
- Состояние — пара $(last, mask)$. $last$ — последняя посещенная вершина, $mask$ — множество посещенных вершин.

Решение задачи

- Можно заметить, что из числа единиц в mask определяется, какой участник бежит в данный момент, и сколько точек ему осталось пробежать.
- Перебираем контрольную точку *new*, которую посетим следующей.
- Делаем переход в $(new, mask|2^{new})$. Если участник еще может бежать, затраченное время — $a_{last,new}$.
- Если должен бежать новый участник, стоимость перехода $a_{last,0} + a_{0,new}$.

Задача I. «Сапсан»

Задача I. «Сапсан»

- Идея задачи — Валерия Петрова
- Подготовка тестов — Дмитрий Филиппов
- Разбор задачи — Дмитрий Филиппов

Постановка задачи

- По данному числу n — количеству мест в вагоне Сапсана — сказать, какое максимальное количество человек туда можно посадить, чтобы ровно у половины людей был сосед
- В каждом ряду Сапсана ровно 2 места.

Решение задачи

- Рядов ровно $\frac{n}{2}$
- На каждом ряду посчитаем количество людей, у которых есть сосед
- Если ряд пустой, их 0, если на ряду 1 человек, их 0, если 2 — 2.
- Если есть a рядов с 2 пассажирами и b рядов с 1 пассажиром, то $b = 2a$
- $a + b \leqslant \frac{n}{2} \Leftrightarrow 3a \leqslant \frac{n}{2} \Leftrightarrow a \leqslant \lfloor \frac{\frac{n}{2}}{3} \rfloor$
- Всего пассажиров $2a + b = 4a$, то есть не более $4 \cdot \lfloor \frac{\frac{n}{2}}{3} \rfloor$

Задача J. «Звезды на погонах»

Задача J. «Звезды на погонах»

- Идея задачи — Григорий Шовкопляс
- Подготовка тестов — Михаил Дворкин
- Разбор задачи — Михаил Дворкин

Постановка задачи

- У всех офицеров должно быть от a до b звезд, причем попарно различные количества.
- Нового офицера с c звездами надо понизить, чтобы правило выполнялось.
- Известно, что для этого у него надо удалить минимум d и максимум e звезд.
- Каково минимальное и максимальное возможное количество офицеров до понижения?

Решение задачи — минимальное кол-во

- Офицеру можно удалить максимум e звезд, то есть оставить минимум $c - e$ звезд.
- Значит, все количества звезд в интервале $[a, c - e)$ точно заняты.
- Можно удалить минимум d звезд, то есть оставить максимум $c - d$ звезд.
- Если $c > b$, то точно занят весь интервал $(c - d, b]$.
- Если $c \leqslant b$, то точно занят весь интервал $(c - d, c)$.
- Ответ: $(c - e - a) + (min(c, b + 1) - (c - d + 1))$

Решение задачи — максимальное кол-во

- Какие количества звезд заведомо свободны?
- Только $c - d$ и $c - e$ — ведь на эти позиции можно понизить офицера.
- Если эти два числа равны, то есть $d = e$, тогда весь интервал $[a, b]$ кроме одной позиции может быть занят, и ответ: $b - a$.
- Если $d \neq e$, то весь интервал без двух позиций может быть занят, и ответ: $b - a - 1$.

Задача K. «Пробки»

Задача K. «Пробки»

- Идея задачи — Евгений Замятин
- Подготовка тестов — Дмитрий Филиппов
- Разбор задачи — Дмитрий Филиппов

Постановка задачи

- Есть дорога из n полос, в i-й полосе стоит c_i машин
- Также дано число k
- В конце дорог стоят светофоры, за каждый зеленый сигнал проезжает сколько-то машин
- Злость водителя автомобиля — число машин, стоящих впереди его
- Найти n чисел k_i, таких, что их сумма равна k, а суммарная злость за все время минимальна.

Решение задачи

- Решение — динамическое программирование
- $dp[n][k]$ — минимальная суммарная злость, которую можно получить, если рассматривать только первые n рядов, а сумма чисел в ответе должна равняться k

Пересчет динамики

- $dp[0][0] = 0$
- $dp[n][k] = \max_{k_1 \leqslant k} dp[n-1][k-k_1] + anger(n, k_1)$ — выбираем, сколько машин будет пропускать за один зеленый сигнал n-й светофор
- $anger(n, x)$ — суммарная злость за все время водителей на n-й полосе, если светофор за один зеленый сигнал будет пропускать x машин

Как посчитать *anger*? Способ 1: формулы

- $cnt = \lfloor \frac{c_n - 1}{x} \rfloor$
- $anger(n, x) = (0 + 1 + \ldots + c_n - x - 1) + (0 + 1 + \ldots + c_n - 2x - 1) + \ldots + (0 + 1 + \ldots + c_n - cnt - 1)$
- $0 + 1 + \ldots + c_n - px - 1 = \frac{(c_n - px - 1)(c_n - px)}{2}$
- $anger(n, x) = \frac{1}{2} \cdot ((c_n - x)(c_n - x - 1) + (c_n - 2x)(c_n - 2x - 1) + \ldots + (c_n - cnt - 1)(c_n - cnt))$
- $(c_n - px - 1)(c_n - px) = c_n^2 - px^2 - px - c_n - 2 \cdot px \cdot c_n$
- Итоговая сумма равна $cnt \cdot c_n^2 - (1^2 + 2^2 + \ldots + cnt^2) - \frac{cnt(cnt-1)}{2} - cnt \cdot c_n - 2 \cdot c_n \cdot \frac{cnt(cnt-1)}{2}$

Как посчитать *anger*? Способ 2: динамика

- Считаем еще одну динамику $anger[c][x]$ — суммарная злость, которая будет, если пропустить c машин, по x за раз
- $anger[c][x] = anger[c-x][x] + \frac{(c-x-1)(c-x)}{2}$
- Если хранить все состояния, получаем превышение лимита по памяти
- Храним только нужные — различных c_i не больше 300
- Используем ассоциативный массив (`std::map`, `java.util.HashMap`)

Асимптотика

- Состояния динамики: $O(n^2)$
- Переход: $O(n)$
- Итого: $O(n^3)$

Возможные ошибки

- Переполнение 32-битного типа

Вопросы?