Математики и программисты

А.ШЕНЬ

Имеются N на вид одинаковых предметов, которые на самом деле нескольких различных типов. Известно, что предметы одного из типов составляют большинство (больше N/2). Требуется указать представителя этого большинства, используя детектор, в который можно (за один раз) загрузить два предмета и определить, одного они типа или разных. Сколько таких проб (сравнений) придется сделать?

Если не экономить и сравнивать попарно все предметы, то понадобится $N(N-1)/2$ сравнений (каждый предмет участвует в $N-1$ сравнениях и в каждом сравнении участвуют два предмета). Оказывается, если действовать с умом, можно сделать число проб пропорциональным N (вместо N^2).

Мы сейчас приведем два доказательства этого факта. Первое (как мы надеемся) должно понравиться математикам, второе – программистам.

Оба рассуждения начинаются с такого замечания: *если мы нашли два предмета разных типов, то их можно выбросить, сохранив большинство*. В самом деле, из них максимум один принадлежит большинству, и после его удаления большинство не пропадет (ведь мы выбросили и предмет другого типа). По тем же причинам два приятеля, которые собираются голосовать за разных кандидатов, могут сговориться не идти на выборы, если они почему-либо верят, что подсчет производится честно и что один из кандидатов пользуется поддержкой более половины голосующих (он потеряет максимум один голос за и минимум один голос против).

Математик продолжит рассуждение так. Пусть сначала N четно. Тогда поделим предметы на пары и сравним их в каждой паре (всего $N/2$ сравнений). Те пары, в которых предметы различны, можно выбросить. После этого остаются пары с равными предметами – и достаточно оставить от каждой по одному, так что задача после $N/2$ сравнений сводится к такой же задаче для (максимум) половинного числа предметов, и можно повторять рассуждение.

Если временно забыть о нечетных N, получится формула

$$T(N) \le N/2 + T(N/2),$$

где $T(n)$ – число проб при поиске представителя большинства среди не более чем n предметов. Отсюда следует, что $T(N) < N$. (Индукция: $T(2) = 0$, поскольку в этом случае больше половины – это все; если $T(N/2) < N/2$, то $T(N/2) < N/2 + N/2 = N$. Например, из 8 предметов после 4 сравнений остаются максимум 4, затем после двух остаются 2, всего 4 + 2 = 6 сравнений.)

Конечно, это рассуждение не имеет смысла, пока мы не разобрались с нечетными N. При нечетном N после выбрасывания пар с неравными предметами остаются несколько пар с равными предметами плюс один непарный предмет.

Например, могут остаться пары (a, a), (b, b), (c, c) и еще d без пары (всего 7 элементов). Мы знаем, что среди этих 7 предметов больше половины (т.е. 4 или более) одинаковых. Поэтому как минимум две пары из трех принадлежат большинству (если только одна, то большинства никак не наберется). Значит, предметы искомого типа составляют большинство среди a, b, c, а про d можно забыть.

Если остается не 7 элементов, а, скажем, 9, то ситуация будет другой. Пусть у нас четыре пары (a, a), (b, b), (c, c), (d, d) и непарный предмет e. Для большинства нужно 5 предметов – достаточно двух пар и e. Поэтому среди a, b, c, d предметы искомого типа могут не быть большинством. Но они должны быть большинством среди a, b, c, d, e, иначе будет 5 элементов другого типа (минимум две пары плюс один). Поэтому в этом случае надо не выбрасывать e, а присоединить его.

Легко сообразить, что один из этих двух приемов всегда сработает: мы либо добавляем непарный предмет, либо нет – с тем расчетом, чтобы на следующем шаге оставалось нечетное число предметов. По индукции можно получить и $T(N) < N$, разобрав два случая: $T(2k) \le k + T(k)$ и $T(2k+1) \le k + T(k+1)$. Первое доказательство закончено.

Это рассуждение порадует математика, но не программиста – который вынужден думать о том, как все эти разные случаи четного и нечетного N и четного и нечетного количества пар запрограммировать и ничего не пропустить. Но есть и другой вариант.

Представим себе, что нас заперли в комнате с детектором, N предметами и тремя большими коробками, где эти предметы могут лежать. Коробки предусмотрительно снабжены надписями: «НЕПРОВЕРЕННЫЕ» (Н), «ОДИНАКОВЫЕ» (О) и «ВЫБРОШЕННЫЕ» (В). Изначально все предметы лежат в коробке «НЕПРОВЕРЕННЫЕ» (что согласуется с ее названием), и нам известно, что более половины из них некоторого одного типа X.

Нам строго-настрого велено соблюдать две заповеди:

- Все предметы в коробке «ОДИНАКОВЫЕ» должны на самом деле быть одного типа (если там вообще что-то есть).
- Предметы типа X должны составлять большинство среди всех невыброшенных (находящихся в коробках «НЕПРОВЕРЕННЫЕ» и «ОДИНАКОВЫЕ»).

(Очевидно, в начальном состоянии оба условия выполнены.)

Когда коробка с надписью «НЕПРОВЕРЕННЫЕ» становится пустой, нас выпускают из комнаты (и это заслуженно, поскольку в этот момент предметы типа X по правилам составляют большинство среди предметов в коробке «ОДИНАКОВЫЕ», и можно взять из нее любой предмет – они все одинаковые).

Что мы можем делать, не нарушая заповедей? Несколько очевидных вещей.

Во-первых, если коробка «ОДИНАКОВЫЕ» пуста, а коробка «НЕПРОВЕРЕННЫЕ» – нет, то можно переложить один предмет из Н в О. (Множество невыброшенных предметов остается тем же самым, так что заповедей мы не нарушим.)

Во-вторых, если обе коробки О и Н непусты, то можно взять по одному предмету и сравнить их. Если они одинаковые, то кладем оба в О; если разные, то выбрасываем (кладем в В) оба. Как мы видели, это не противоречит заповедям.

Остается заметить, что

- если есть непроверенные предметы, то одно из действий заведомо выполнимо;
- любое из действий уменьшает число непроверенных на 1, при этом детектор используется не более одного раза.

Следовательно, после N шагов коробка Н опустеет и нас выпустят. При этом детектор будет использован не более N раз (на самом деле меньше, так как первый шаг – перекладывание).

Соответствующая программа (совсем простая) приведена в книжке «Программирование: теоремы и задачи» (М.: МЦНМО, 2-е изд., 2004; см. также сайт www/mccme.ru/freebooks). Программирование первого решения явно потребовало бы существенно больших усилий.

На этом примере можно немного пофилософствовать о разнице между математическим и программистским взглядом на вещи. Пусть задача X сводится к множеству однородных простых задач $X_1, \ldots, X_n$. Математик обычно разбирает одну из них и пишет «$X_2, \ldots, X_n$ решаются аналогичным способом и оставляются читателю в качестве упражнений». Напротив, программист понимает, что никто за него программировать не будет, а программирование большого количества даже и очень простых задач утомительно, и лучше найти другой подход, избегающий разбора многочисленных случаев.

КВАНТЫ ИНТЕРНЕТА

На сколько частей делят пространство плоскости граней додекаэдра?

(Начало см. на 2-й странице обложки)

На Межрегиональной заочной математической олимпиаде 2008 года предлагались задачи:

Найти число частей, на которые продолжения граней тетраэдра разбивают пространство (задача 5, 7 класс).

Найти число частей, на которые продолжения граней куба разбивают пространство (задача 3, 6 класс).

Обе эти задачи решаются простым подсчетом. В первом случае ответ 15; во втором 27. Но возникает вопрос, а на сколько частей разбивают пространство продолжения граней других правильных многогранников: октаэдра, додекаэдра и икосаэдра. Подсчет в этом случае уже сложнее, но задача вполне разрешима! Рассмотрим эту задачу для додекаэдра.

Есть два способа подсчета числа частей, на которые продолжения граней додекаэдра разбивают пространство, не требующих выхода из плоскости.

Первый способ. Разобьем продолжения граней додекаэдра на 6 пар параллельных плоскостей. Теперь будем последовательно добавлять эти пары плоскостей и смотреть, сколько частей получается в результате. При добавлении пары плоскостей количество частей увеличивается на число частей, которые данные плоскости пересекают.

В начале есть одна часть (все пространство). Каждая из плоскостей первой пары пересекает 1 часть. Поэтому после добавления одной пары плоскостей число частей будет равно трем. Аналогично, вторая пара плоскостей пересекает по 3 части, а третья пара – по 9 частей. Поэтому после добавления трех пар плоскостей число частей станет равным 27. Чтобы посчитать число частей, добавляющееся на четвертом, пятом и шестом шаге, рассмотрим чертеж, на котором изображены пересечения плоскости одной из граней продолжениями других плоскостей (рис.1, 2). На рисунке 1 одинаковым цветом изображены пересечения данной плоскости с продолжениями пар параллельных граней. Здесь любая тройка пар параллельных прямых разбивает плоскость на 17 частей. Следовательно, четвертая пара плоскостей пересекает по 17 частей. Таким образом, после проведения четырех пар плоскостей количество частей станет равным $27 + 17 \cdot 2 = 61$. Аналогично, любая четверка пар параллельных прямых на рисунке разбивает плоскость на 26 частей. Поэтому после проведения пяти пар плоскостей количество частей станет равным $61 + 2 \cdot 26 = 113$. Наконец, все пять пар разбивают плоскость на 36 частей (рис.2). Поэтому после добавления последней (шестой) пары плоскостей число пространственных частей станет равным $113 + 36 \cdot 2 = 185$.

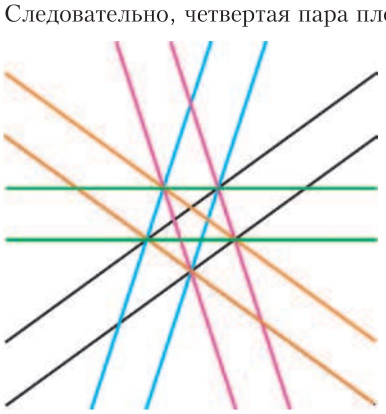

Рис. 1

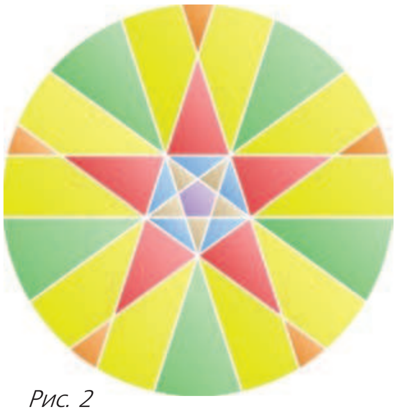

Рис. 2

Второй способ. Хорошо известна формула Эйлера для плоского графа. Вот один из ее частных случаев:

Пусть на плоскости проведено несколько прямых. Обозначим через В число вершин – точек пересечения нескольких прямых; Р – число ребер – отрезков между парами вершин и лучей, выходящих из вершины; Г – число граней – многоугольников, на которые данные прямые разбили плоскость. Тогда Г – Р + В = 1.

Менее известно, что аналогичная формула есть и в пространственном случае (в действительности такая формула есть и для пространства произвольной размерности):

Пусть в пространстве проведено несколько плоскостей. Обозначим через В число вершин – точек пересечения нескольких прямых; Р – число ребер – отрезков между парами вершин и лучей, выходящих из вершины; Г – число граней – многоугольников, которые в данных плоскостях высекаются другими плоскостями; Ч – число пространственных частей, на которое пространство разбивается данными плоскостями. Тогда Ч – Г + Р – В = 1.

Таким образом, количество частей, на которые данное множество плоскостей разбивает пространство, может быть вычислено по формуле: $Ч = Г - Р + В + 1$.

Как видно из рисунка 2, в каждой из плоскостей содержится 36 «*граней*». Количество плоскостей равно числу граней додекаэдра, т.е. 12. Поэтому полное число «*граней*» пространственного графа равно $Г = 12 \cdot 36 = 432$.

Аналогично можно посчитать количество «*ребер*». Каждая плоскость содержит 10 прямых, разбитых на 5 «*ребер*», а все 12 плоскостей – на $5 \cdot 10 \cdot 12 = 600$ «*ребер*». Но каждое «*ребро*» принадлежит двум плоскостям. Следовательно, количество «ребер» вдвое меньше и равно $Р = 600/2 = 300$.

Несколько сложнее посчитать количество «*вершин*». В каждой плоскости есть 5 «*вершин*», являющихся пересечением четырех прямых (и, следовательно, пяти плоскостей) и 10 «*вершин*», являющихся пересечением двух прямых (и, следовательно, трех плоскостей). Можно считать, что в каждой плоскости находится 5 «*пятых частей вершин*» и 10 «*третьих частей вершин*». Значит, общее число «*вершин*» данного пространственного графа равно $В = (5/5 + 10/3) \cdot 12 = 52$. Следовательно, $Ч = 432 - 300 + 52 + 1 = 185$.

Л.Оридорога, М.Панов