Дискретная математика **Итоговый экзамен 2014-2015, решения**

1. Существуют ли такие множества $X, Y \subseteq \mathbb{N}$, что X разрешимо, $X \cup Y$ разрешимо, а Y не разрешимо? Дайте обоснованный ответ.

Решение. Возьмем $X = \mathbb{N}$, а в качестве Y – неразрешимое множество (в рамках курса мы доказывали существование такого множества). Тогда X разрешимо, $X \cup Y = \mathbb{N}$ разрешимо, а Y не разрешимо. □

2. Сколько существует различных простых неориентированных графов на 10 вершинах $v_1, \ldots, v_{10}$ с 4 ребрами, таких, что все ребра сходятся в одной вершине (т.е. существует вершина, которая является концом всех четырех ребер графа)? Ответом должно быть число. Ответ должен быть обоснован.

Решение. В таком графе степень одной вершины будет равна 4 (вершина, являющаяся концом все 4 ребер), степени других 4 вершин будут равны 1 (другие концы 4 ребер; эти вершины не совпадают, поскольку в простом графе нет кратных ребер), а степени остальных вершин равны 0.

Вершину степени 4 из 10 вершин графа можно выбрать десятью способами. Для всякого выбора вершины степени 4 из оставшихся 9 вершин можно выбрать 4 вершины степени один $\binom{9}{4}$ способами. Каждому выбору таких вершин соответствует некоторый граф из условия задачи, и наоборот, для всякому графу соответствует некоторый набор вершин.

Следовательно, всего таких различных графов

$$10 \cdot \binom{9}{4} = 10\frac{9 \cdot 8 \cdot 7 \cdot 6}{4 \cdot 3 \cdot 2} = 10 \cdot (9 \cdot 2 \cdot 7) = 1260.$$

□

3. Существует ли главная (гёделева) универсальная вычислимая функция $U \colon \mathbb{N} \times \mathbb{N} \to \mathbb{N}$, такая что множество номеров $\{n \mid U_n(x) = U(n, x)$ — нигде не определенная функция$\}$ совпадает с множеством четных чисел? Обоснуйте ответ.

Решение. По теореме Райса-Успенского для любого нетривиального свойства функций $\mathcal{A}$ множество $\{n \mid U_n \in \mathcal{A}\}$ – неразрешимо. Свойство $\mathcal{A} = \{f_\emptyset\}$, где $f_\emptyset \colon \mathbb{N} \to \mathbb{N}$ – нигде не определенная функция, нетривиально, так как есть как функция, обладающая свойством (функция $f_\emptyset$, так и функция им не обладающая (например, $f(x) \equiv 0$). Следовательно, множество $\{n \mid U_n = f_\emptyset\}$, указанное в задаче, неразрешимо. С другой стороны, множество четных чисел разрешимо, а значит эти множества не могут совпадать.

□

4. Пусть дана система подстановок в некотором конечном алфавите, в которой для каждой подстановки $L \to R$ в ее левой части L стоит буква, а в ее правой части R стоит непустое слово. Докажите, что задача достижимости в графе, заданном этой системой подстановок, разрешима. Необязательно (и даже нежелательно) приводить описание МТ, решающей эту задачу, или код на любом другом языке программирования. Обязательно обоснование корректности алгоритма..

Решение. Заметим, что в такой системе подстановок длина слов при преобразовании не уменьшается. Следовательно, путь из слова u в слово v может проходить только по словам длины не большей, чем длина слова v (обозначим эту длину через n). Следовательно, если такой путь есть, то он есть и в индуцированном подграфе, порожденном словами длины не больше n. Этот подграф конечный, а нам известны алгоритмы решения задачи достижимости в конечных графах.

Для полноты изложения, опишем полный алгоритм, разрешающий задачу достижимости. Пусть на вход нам даны слова u и v и требуется проверить, есть ли путь из u в v в заданном графе. Для этого достаточно проверить, есть ли он в подграфе $G = (V, E)$, порожденном словами длины не больше n, где n длина поданного на вход слова v. Заметим, что граф G конечный, так что теперь нам нужно решить задачу достижимости в конечном графе. Мы построим последовательно множества $S_0, S_1, S_2, \ldots, S_k \subseteq V$, множество S_i состоит из всех вершин графа, достижимых из u путем длины не больше i. Множество S_0 состоит из одной вершины u. Пусть уже построено множество S_i, опишем, как строить множество S_{i+1}. Во-первых, в него входят все элементы множества S_i. Далее, мы проходим по всем элементам множества S_i (конечное число), по всем правилам подстановок (конечное число) и проверяем, применимо ли данное правило к данному слову. Если правило применимо, проверяем, принадлежит ли результат применения правила множеству уже найденный слов. Если не принадлежит, добавляем это слово в S_{i+1}, если же принадлежит, переходим к следующей паре (слово из S_i, правило). После построения очередного множества S_{i+1} мы проверяем, совпадает ли оно с S_i. Если совпадает, процесс построения последовательности множеств заканчивается, если же не совпадает, переходим к построению S_{i+2}. Нетрудно видеть (и можно формально доказать по индукции), что множество S_i состоит в точности из тех вершин G, которые достижимы из u путем длины не более i. Если для кого-то i оказалось, что $S_i = S_{i+1}$, это означает, что все ребра, выходящие из S_i, ведут также внутрь S_i. А значит, мы построили множество всех вершин, достижимых из вершины u. Заметим, что поскольку множество V конечно, рано или поздно случится, что $S_i = S_{i+1}$, и процесс закончится.

Когда этот процесс завершен, остается лишь проверить, лежит ли вершина v в множестве S_k. Если лежит, то она достижима из u, если нет, то не достижима.

Нетрудно видеть, что все шаги описанного процесса можно выполнить алгоритмически, например, на машине Тьюринга.

$\square$

5. В графе G на множестве вершин $\{0, 1, \ldots, 66\}$ ребром соединены пары вершин x, y, для которых выполняется $13(x - y) \equiv \pm 4 \pmod{67}$. Является ли граф G связным? Обоснуйте ответ.

Решение. Поскольку 67 простое число, всякий вычет по модулю 67 обратим. Обозначим обратный вычет к 13 через a. Тогда $(x - y) \equiv \pm 4 \cdot a \pmod{67}$, а значит из всякой вершины x есть ребро в вершины $x \pm 4 \cdot a \pmod{67}$, то есть из всякого вычета x можно сдвинуться на $4 \cdot a \pmod{67}$ в обе стороны. Обозначим $4 \cdot a \pmod{67}$ через d. Далее, поскольку, опять же, 67 простое число, d обратим по модулю 67, обозначим обратный к нему через e, то есть $d \cdot e \equiv 1 \pmod{67}$. Тогда из всякой вершины x мы за e шагов по указанным выше ребрам попадем в вершину $x + e \cdot d \equiv x + 1 \pmod{67}$, то есть из каждой вершины можно попасть в следующую. Повторяя этот процесс, из любой вершины можно дойти до любой другой, а значит граф связен.

$\square$

6. Пусть дан неубывающий числовой массив $a[1], \ldots, a[n]$, то есть $a[i] \leq a[i+1]$ для всякого i. Разрешается за один ход сравнить любые два элемента массива. Требуется проверить, есть ли в массиве равные элементы. Найдите минимальное количество вопросов, достаточное для такой проверки. Дайте обоснованный ответ.

Решение. Во-первых, заметим, что достаточно сделать $n - 1$ сравнение. Действительно, для всякого $i = 1, \ldots, n-1$ сравним элементы $a[i]$ и $a[i+1]$. Если какие-то два из них совпадают, то не все элементы массива различны. Если же все эти элементы различны, то массив является возрастающим и никакие два его элемента не совпадают.

Докажем теперь, что меньшим числом вопросов обойтись нельзя. Для этого опишем стратегию противника, вынуждающую задать не меньше $n - 1$ вопросов. Стратегия довольно проста: для всякого запроса о числах $a[i]$ и $a[j]$, где $i < j$, противник будет отвечать, что $a[i] < a[j]$.

Предположим, что сделано не больше $n-2$ сравнений. Рассмотрим пары $(a[1], a[2]), (a[2], a[3]), \ldots, (a[n-1], a[n])$. Таких пар $(n-1)$ штука, значит хотя бы для одной из них сравнение произведено не было, пусть это пара $(a[k], a[k+1])$. Теперь рассмотрим два различных исходных неубывающих массива. Первый будет возрастающим, то есть все числа в нем различны. Во втором числа $a[k]$ и $a[k+1]$ совпадают, а все остальные числа различны. На обоих массивах все сделанные сравнения дают одни и те же результаты (и эти результаты совпадают с ответами противника). При этом результат вычисления на этих массивах должен быть разный, так как в одном массиве все элементы различны, а в другом есть совпадающие. Следовательно, алгоритм, делающий не более $n-2$ сравнений, даст неправильный ответ на одном из построенных входов.

$\square$

7. Таблица из n столбцов и $n+1$ строки заполняется числами ± 1 по следующему правилу. Первая строка таблицы полностью заполняется числами 1. Строка с номером $i+1$ получается из строки с номером i так: в одном случайно и равновероятно выбранном столбце значение меняется на противоположное, а в остальных столбцах значения сохраняются. Случайная величина X равна сумме чисел в последней $(n+1)$-й строке таблицы. Найдите $\lim_{n\to\infty} E[X]/n$. Дайте обоснованный ответ.

Решение. Рассмотрим следующие случайные величины X_j, для всех $j = 1, \ldots, n$: $X_i = 1$, если j-ая ячейка последней строки равна 1, и $X_j = -1$ иначе. Нетрудно видеть, что $X = X_1 + \ldots + X_n$. В силу линейности математического ожидания $E[X] = \sum_j E[X_j]$.

Найдем $E[X_j]$ для произвольного j. Для этого найдем $Pr[X_j = 1]$ и $Pr[X_j = -1]$. Заметим, что $X_j = 1$ тогда и только тогда, когда j-ая координата строки менялась четное число раз. Вероятность того, что она менялась ровно i раз равна

$$\binom{n}{i} \cdot \left(\frac{1}{n}\right)^i \cdot \left(\frac{n-1}{n}\right)^{n-i},$$

где первый множитель соответствует выбору i шагов, на которых менялась j-ая координата, второй множитель соответствует тому, что значение j-ой координаты изменилось на выбранных шагах, а третий – тому, что значение не менялось на остальных шагах.

Таким образом,

$$Pr[X_j = 1] = \sum_{i \text{ — четное}} \binom{n}{i} \cdot \left(\frac{1}{n}\right)^i \cdot \left(\frac{n-1}{n}\right)^{n-i}.$$

Аналогично,

$$Pr[X_j = -1] = \sum_{i \text{ — нечетное}} \binom{n}{i} \cdot \left(\frac{1}{n}\right)^i \cdot \left(\frac{n-1}{n}\right)^{n-i}.$$

Тогда

$$E[X_j] = 1 \cdot Pr[X_j = 1] + (-1) \cdot Pr[X_j = 1] = \sum_{i=0}^{n} \binom{n}{i} \cdot \left(-\frac{1}{n}\right)^i \cdot \left(\frac{n-1}{n}\right)^{n-i} = \left(\frac{n-1}{n} - \frac{1}{n}\right)^n = \left(1 - \frac{2}{n}\right)^n.$$

Следовательно,

$$E[X] = \sum_i E[X_j] = n \cdot \left(1 - \frac{2}{n}\right)^n.$$

Наконец,

$$\lim_{n\to\infty} E[X]/n = \lim_{n\to\infty} \left(1 - \frac{2}{n}\right)^n = \lim_{n\to\infty} \left(\left(1 - \frac{2}{n}\right)^{n/2}\right)^2 = e^{-2}.$$

$\square$

8. Докажите, что для всех достаточно больших n существует монотонная булева функция $f\colon \{0,1\}^n \to \{0,1\}$, которую нельзя вычислить схемой размера меньше n^{100}. (Булева функция f называется монотонной, если из неравенств $x_i \le y_i$ для всех i следует неравенство $f(x_1, \ldots, x_n) \le f(y_1, \ldots, y_n)$.)

Решение. В лекциях доказывалось аналогичное утверждение, но для произвольных (а не только монотонных функций). В частности, в этом доказательстве доказывалось, что схему размера не больше S можно описать $S \cdot (2 + 2\log_2 S)$ битами, а значит всего таких схем не больше $2^{S\cdot(2+2\log_2 S)}$.

Проведем доказательство аналогично доказательству из лекции. Для этого нам нужно оценить снизу количество монотонных функций вида $f\colon \{0,1\}^n \to \{0,1\}$. Для этого рассмотрим семейство монотонных функции специального вида, а именно такие функции, что $f(x_1,\ldots,x_n) = 0$, если $\sum_i x_i < \lfloor n/2 \rfloor$, и $f(x_1,\ldots,x_n) = 1$, если $\sum_i x_i > \lfloor n/2 \rfloor$. На наборах, для которых $\sum_i x_i = \lfloor n/2 \rfloor$, эти функции могут принимать любые значения. Нетрудно видеть, что всякая такая функция монотонна. Действительно, если для всех i выполняется $x_i \le y_i$, то либо $(x_1,\ldots,x_n) = (y_1,\ldots,y_n)$, либо $\sum_i x_i < \sum_i y_i$. В обоих случаях $f(x_1,\ldots,x_n) \le f(y_1,\ldots,y_n)$.

Оценим количество таких функций. Количество наборов $(x_1,\ldots,x_n)$, таких что $\sum_i x_i = \lfloor n/2 \rfloor$ равно $\binom{n}{\lfloor n/2 \rfloor}$. На каждом из этих наборов функция принимает одно из двух значений, следовательно всего таких функций $2^{\binom{n}{\lfloor n/2 \rfloor}}$.

Теперь нам нужно оценить величину $\binom{n}{\lfloor n/2 \rfloor}$. В лекции мы доказывали, что $\binom{n}{\lfloor n/2 \rfloor} \ge 2^n/(n+1)$ (количество биномиальных коэффициентов вида $\binom{n}{i}$ равно $(n+1)$, их сумма равна 2^n, и наш среди них максимален).

Таким образом, мы получаем, что булевых схем размера не больше S не больше $2^{S\cdot(2+2\log_2 S)}$, а монотонных функций от n переменных не меньше $2^{2^n/(n+1)}$. Заметим, что если $S = n^{100}$, то для всех достаточно больших n

$$2^{n^{100}\cdot(2+200\log_2 n)} < 2^{2^n/(n+1)},$$

а значит не всякую монотонную функцию от n переменных можно вычислить схемой размера не больше n^{100}.

$\square$