Содержание

1 Семинар 5

1.1 Задача 1

Сколько существует способов переставить буквы в слове ОБОРОНОСПОСОБНОСТЬ так, чтобы никакие две буквы О не стояли рядом?

Слово ОБОРОНОСПОСОБНОСТЬ содержит 7 букв «О» и 11 букв не «О», а именно: 2 буквы Б, 2 буквы Н, 3 буквы С, остальные буквы по одному разу. Тогда ясно, что способов переставить «не О» $\frac{11!}{2!\cdot 2!\cdot 3!}$

Теперь давайте считать все буквы «не О» перегородками, а буквы «О» – шарами. Тогда нужно в каждый промежуток между перегородками (а также до и после перегородок) поставить не более одного шара. Или другими словами, для каждого шара выбрать промежуток, куда мы его поставим, но при этом повторяться эти промежутки не могут.

Перегородок 11 штук, поэтому подходящих промежуток 12 (один слева от первой перегородки, 10 между перегородками, один справа от последней перегородки). Тогда для первого шара я могу выбрать один из 12 промежутков, для второго – один из 11, ..., для седьмого – один из шести. Но вообще говоря, я не различаю буквы О (шары) между собой, поэтому ещё нужно делить на 7! Получаем C_{12}^7.

Не забываем домножить на количество способов переставить буквы не О и получаем ответ.

Ответ: $C_{12}^7 \cdot \frac{11!}{2!\cdot 2!\cdot 3!}$

1.2 Задача 2

Сколькими способами можно разделить 15 одинаковых монет между 7 нумизматами так, чтобы каждому досталось хотя бы по монете?

Давайте всем ребятам сразу дадим по монетке, чтобы никого не обидеть. Тогда останется 8 монеток, которые нужно распределить между 7 ребятами.

Будем считать, что у нас 6 перегородок (на 1 меньше, чем число ребят), а 8 монеток – шарами. Ну и все что нужно сделать – это любыми способами расставить 8 шаров и 6 перегородок. Ясно, что число способов это сделать – C_{14}^6

Ответ: C_{14}^6

1.3 Задача 3

Давайте сразу дадим всем числам по 5. Тогда останется раздать 75 между 5 числами. Поставим 4 перегородки и 75 шариков. Думаю, понятно, что всё, что идёт до первой перегородки – идет в первое число, всё что идет между первой и второй перегородкой – идёт во вторoе число ну и так далее.

79 мест и надо просто расставить 4 перегородки, на эти 79 мест. Итог: C_{79}^4

Ответ: C_{79}^4

1.4 Задача 4

Сколько существует способов представить число n в виде суммы натуральных слагаемых, если способы, отличающиеся лишь порядком слагаемых, считаются различными?

Ясно, что слагаемых не более, чем n. Пусть слагаемых k штук. Аналогично предыдущей задаче можно про это думать так: каждому из k слагаемых мы раздали по единице. А оставшиеся $n-k$ можно распределять как угодно. Для этого нужно лишь расставить $n-k$ шаров и $k-1$ перегородок. Ясно, что всего объектов n, а расстановка $k-1$ перегородок однозначно задаёт всё. Поэтому число способов это сделать C_{n-1}^{k-1}.

Тогда всё что нужно сделать это взять сумму

$$\sum_{k=1}^{n} C_{n-1}^{k-1} = \sum_{k=0}^{n-1} C_{n-1}^{k} = 2^{n-1}$$

Ответ: 2^{n-1}

1.5 Задача 5

Докажи тождество, не применяя формулу для количества сочетаний:

$$C_n^k \cdot C_k^m = C_n^m \cdot C_{n-m}^{k-m}$$

Доказательство. Слева написано следующее: сначала из n человек выберем k крутых ребят, а потом из k крутых ребят выберем m мега-крутых ребят.

Справа написано следующее: сначала из n человек выберем m мега-крутых ребят, а потом из оставшихся $n-m$ выберем $k-m$ просто крутых ребят.

Ясно, что мы посчитали одно и то же двумя способами, что доказывает это тождество □

1.6 Задача 6

Докажи тождество, не применяя формулу для количества сочетаний:

$$C_{n+1}^{k+1} = C_n^k + C_{n-1}^k + C_{n-2}^k + \ldots + C_k^k$$

Доказательство. Заметим, что $C_{n+1}^{k+1} = C_n^k + C_n^{k+1}$. (Либо мы берем $n+1$ элемент и тогда среди оставшихся n нужно выбрать ещё k, либо мы не берем $n+1$ элемент и тогда среди оставшихся всё ещё нужно выбрать $k+1$).

Аналогично можно расписать последнее слагаемое в этой сумме:

$$C_{n+1}^{k+1} = C_n^k + C_n^{k+1} = C_n^k + C_{n-1}^k + C_{n-1}^{k+1}$$

Продолжаем действовать аналогично и в итоге приходим к такой сумме:

$$C_{n+1}^{k+1} = C_n^k + C_{n-1}^k + C_{n-2}^k + \ldots + C_k^k$$

□

1.7 Задача 7

а) В столовой университета продается n различных блюд. Илья каждый день ходит в столовую и заказывает некоторый набор блюд, но со странным ограничением: ни в какие два дня наборы блюд, которые он заказал, не должны повторяться. Какое максимальное количество дней Илья сможет так ходить в столовую и какое суммарное количество блюд он закажет за это время?

б) Из n человек нужно выбрать команду и назначить в ней капитана. Команда может состоять и из одного человека, тогда он будет капитаном.) Сколько существует способов это сделать?

в) Используя пункты (а) и (б), докажи тождество:

$$C_n^1 + 2C_n^2 + \ldots + nC_n^n = n \cdot 2^{n-1}$$

1.7.1 Пункт А

Ясно, что Илья сможет ходить в столовую 2^n дней, так как на наборе из n элементов есть 2^n уникальных подмножеств (каждый элемент либо берем, либо не берем).

Какое число блюд Илья съест за это время? Фиксируем конкретное блюдо. Тогда на оставшихся $n-1$ типах блюд он мог заказать всё что угодно (по 2 варианта для каждого типа блюда: берём в заказ, или не берём. Тогда для фиксированного типа блюда Илья съест их 2^{n-1}.

Но так как различных типов блюд всего n, то всего блюд будет съедено $n \cdot 2^{n-1}$

1.7.2 Пункт Б

Можно взять в команду одного человека и из этого одного выбрать капитана. Можно взять в команду двух людей и из этих двоих выбрать одного капитаном. Можно взять в команду k людей и из этих k выбрать одного капитаном.

Ясно, что итоговое число способов будет равно:

$$C_n^1 + 2C_n^2 + \ldots + nC_n^n$$

1.7.3 Пункт В

Доказательство. Надо лишь понять, почему пункты а и б решают одну и ту же задачу. Ну, например, увидим задачу про блюда в задаче про капитанов. Давайте сначала выберем одного из n людей капитаном, а затем из оставшихся $n-1$ людей каждого мы либо берём, либо не берём. Получается, что всего способов набрать команду таким образом ровно $n \cdot 2^{n-1}$. Это объясняет тождество

$$C_n^1 + 2C_n^2 + \ldots + nC_n^n = n \cdot 2^{n-1}$$

В задаче про блюда это можно увидеть следующим образом: если Илья заказал набор из k блюд, то он (удивительно) съест k блюд. Способов выбрать k блюд из n всего C_n^k. Поэтому Илья съест следующее число блюд:

$$C_n^1 + 2C_n^2 + \ldots + nC_n^n$$

Ну а ранее мы уже показали, что эту же величину можно посчитать и как $n \cdot 2^{n-1}$ $\square$

1.8 Задача 8

В предыдущих задачах, а также в лонгриде мы доказывали нижеперечисленные тождества комбинаторным способом. Докажи эти тождества, используя бином Ньютона:

а) $C_n^0 + C_n^2 + C_n^4 + \cdots = C_n^1 + C_n^3 + C_n^5 + \cdots$

б) $(C_n^0)^2 + (C_n^1)^2 + \cdots + (C_n^n)^2 = C_{2n}^n$

в) $C_n^1 + 2C_n^2 + 3C_n^3 + \cdots + nC_n^n = n \cdot 2^{n-1}$

1.8.1 Пункт А

Доказательство. Нетрудно заметить, что $(1-1)^n = C_n^n \cdot 1^n \cdot (-1)^0 + C_n^{n-1} \cdot 1^{n-1} \cdot (-1)^1 + \ldots + C_n^0 \cdot 1^0 \cdot (-1)^n$.

Слева написан ноль, поэтому давайте перенесем слагаемые, у которых цэшка написана с минусом влево (то есть у которых минус единица в четной степени). Тогда получим требуемое. $\square$

1.8.2 Пункт Б

Доказательство. Нетрудно заметить, что $(1+x)^n \cdot (1+x)^n = (1+x)^{2n}$

Раскроем по биному скобки слева, а именно посчитаем коэффициент при x^n:

$$(1+x)^n \cdot (1+x)^n = \sum_{k=0}^{n} = C_n^k \cdot C_n^{n-k} = \sum_{k=0}^{n} C_n^k \cdot C_n^k = \sum_{k=0}^{n} (C_n^k)^2$$

А скобки справа раскрываются совсем просто, это просто C_{2n}^n. Это доказывает требуемое. $\square$

1.8.3 Пункт В

Доказательство.

$$(x+1)^n = C_n^0 + C_n^1 x + \ldots + C_n^n x^n$$

Продифференцируем по x:

$$n \cdot (x+1)^{n-1} = C_n^1 + 2C_n^2 \cdot x + \ldots + nC_n^n x^{n-1}$$

Подставим $x = 1$:

$$n \cdot 2^{n-1} = C_n^1 + 2C_n^2 \cdot + \ldots + nC_n^n$$

$\square$

1.9 Задача 9

Докажи эти тождества, используя треугольник Паскаля:

а) $C_n^0 + C_n^2 + C_n^4 + \cdots = C_n^1 + C_n^3 + C_n^5 + \cdots$

б) $(C_n^0)^2 + (C_n^1)^2 + \cdots + (C_n^n)^2 = C_{2n}^n$

1.9.1 Пункт А

Доказательство. Если n четно, то достаточно воспользоваться симметрией треугольника Паскаля.

Если же n нечетно, то можно воспользоваться тем, что каждое число в треугольнике Паскаля получено из суммы двух чисел выше него и свести задачу к четному случаю, не хочу это подробно расписывать. $\square$

1.9.2 Пункт Б

Доказательство. Интерпретация правой части такая: оказывается, что C_{2n}^n - количество путей из верха треугольника Паскаля до C_{2n}^n (и вообще число путей до C_n^k в треугольнике Паскаля равно C_n^k, так как по сути на каждом «шаге» из вершины треугольника вы можете пойти либо вправо, либо влево, следовательно, достаточно выбрать k раз, когда вы пойдете вправо)

А что написано в левой части? Смотрите: чтобы дойти до точки C_{2n}^n нужно сделать n шагов влево и n шагов вправо.

Тогда просто нужно выбрать стрелки влево, оставшиеся - будут стрелки вправо.

Но n стрелок влево можно выбрать так: из первых n выбираем 0 стрелок влево, из последних n выбираем n стрелок влево. Либо из первых n стрелок выбираем 1 стрелку влево, из последних n выбираем $n-1$ стрелку влево. Ну и так далее. Итог: $C_n^0 C_n^n + C_n^1 C_n^{n-1} + \ldots C_n^n C_n^0$.

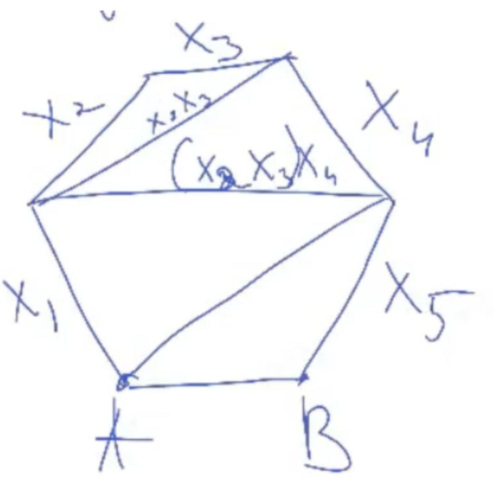
Вот и получается, что мы посчитали одно и то же двумя способами. □

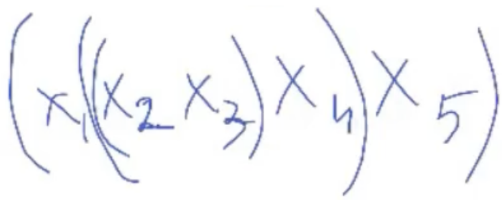
1.10 Задача 10

Сколькими способами можно разрезать правильный n-угольник на треугольники, проводя непересекающиеся диагонали?

Утверждение. *Для правильного n-угольника количество его триангуляций равно c_{n-2} (c_k - кол-во правильных скобочных последовательный из $2k$ скобок).*

Доказательство. Давайте просто покажем, что между триангуляциями и правильными скобочными последовательностями (ПСП) есть биекция. Это объяснит наше утверждение.





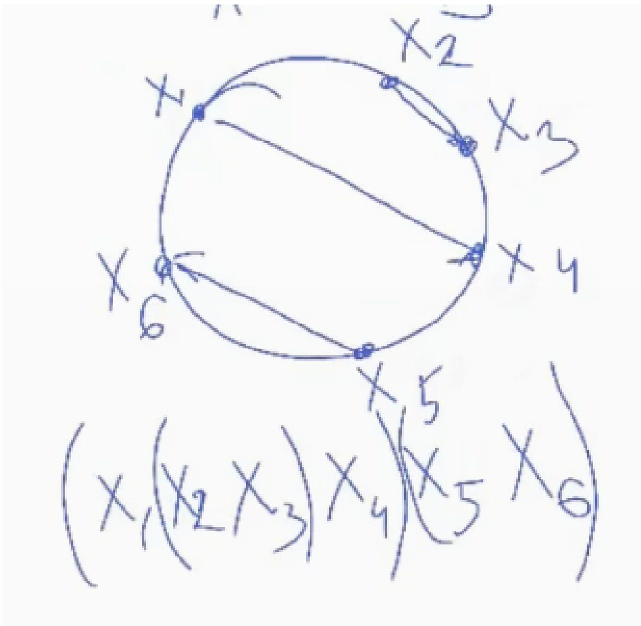
Думаю, по этой картинке понятно, как устроена биекция и как получать картинку из ПСП и наоборот. Не хочется подробно расписывать =) □

1.11 Задача 11

На окружности отмечено $2n$ различных точек. Сколькими способами можно провести n попарно непересекающихся хорд, соединяющих эти точки, так, чтобы каждая точка была концом ровно одной из проведенных хорд?

Утверждение. *Для правильного $2n$ различных точек число искомых способов равно c_n*

Доказательство. Давайте просто покажем, что между выбором хорд и правильными скобочными последовательностями (ПСП) есть биекция.



Опять же не хочу ничего подробно расписывать, картинка многое объясняет, ну, разве что стоит немного аккуратно поговорить про то, что хорды не пересекаются, но я не хочу этим заниматься. □

2 Семинар 6

2.1 Задача 1

Докажи, что клетчатый квадрат размера $2^n \times 2^n$ без любой клетки можно разрезать на уголки, состоящие из трёх клеток.

Доказательство. База индукции при $n = 1$ очевидна.

Предположим, что я умею разрезать квадратик $2^n \times 2^n$ без любой клетки на уголки, состоящие из трёх клеток, и мне дали любой квадрат $2^{n+1} \times 2^{n+1}$ без любой клетки.

Первое, что нужно сделать – разрезать его на 4 квадрата $2^n \times 2^n$ как показано на рисунке:

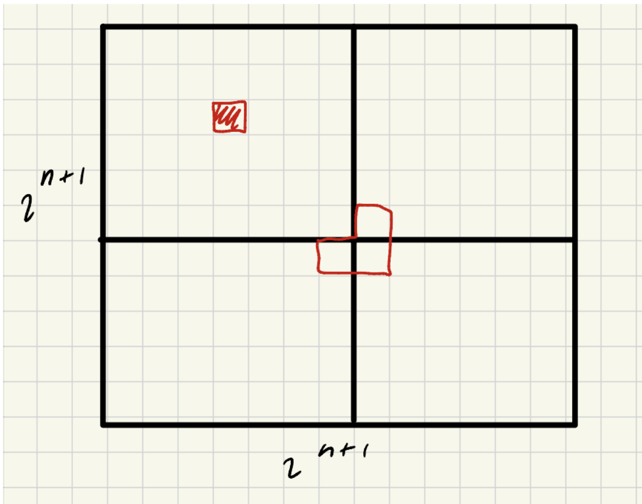


Рис. 1: Заштрихованная красная клетка отсутствует в квадрате $2^{n+1} \times 2^{n+1}$

Тогда отсутствующая клетка попадает в один из этих квадратов. Но тогда в оставшихся трёх я могу вырезать по клетке так, чтобы в итоге образовался уголок (аналогично картинке). Тогда получается 4 квадрата $2^n \times 2^n$. По предположению индукции я умею разрезать квадрат $2^n \times 2^n$ без любой клетки на уголки, состоящие из трёх клеток. Поэтому я смогу разрезать каждый из четырех квадратов на уголки. Значит, и весь квадрат $2^{n+1} \times 2^{n+1}$ без одной клетки тоже побьётся на уголки. Переход доказан. □

2.2 Задача 2

На плоскости провели n прямых так, что любые две проведённые прямые пересекаются, и никакие три прямые не пересекаются в одной точке. На сколько частей прямые поделят плоскость?

Посмотрим, как устроен ответ при маленьких n:

Кол-во прямых	**Кол-во частей плоскости**
0	1
1	2
2	4
3	7
4	11

Возникает гипотеза, что ответ вычисляется по такой формуле:

$$\frac{n(n+1)}{2} + 1$$

Доказательство. Докажем это по индукции:

База индукции. при $n = 0$ очевидна.

Предположение индукции. Предположим, что n прямых делят плоскость на $\frac{n(n+1)}{2} + 1$ частей

Доказательство перехода индукции. Хотим доказать, что $n+1$ прямая делит плоскость на $\frac{(n+1)(n+2)}{2} + 1$ частей.

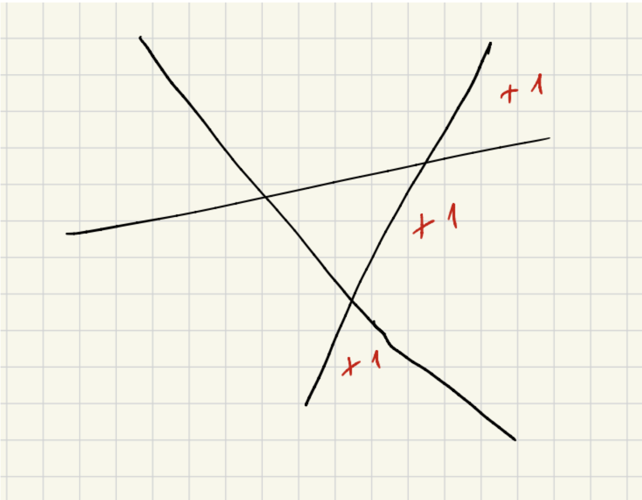
Давайте выкинем из конструкции с $n+1$ прямой одну прямую, проверим, что она является корректной конструкцией (любые две проведённые прямые пересекаются, и никакие три прямые не пересекаются в одной точке, далее такие конструкции называются конструкцией общего положения) из условия с n прямыми, а затем убедимся в верности утверждения для $n + 1$.

Важно: мы делаем именно так, а не прибавляем к n прямых ещё одну потому что не ясно, почему это не испортит конструкцию из условия.

Действительно: если мы из конструкции с $n+1$ прямых общего положения выкидываем одну прямую, то получаем конструкцию общего положения из n прямых. Следовательно, в ней есть ровно $\frac{n(n+1)}{2} + 1$

Давайте вернем выкинутую прямую и проверим, что с ней тоже утверждение.

По условию любые 2 прямые пересекаются, следовательно, $n + 1$ прямая пересекается с n прямыми. Они разбивают эту прямую на $n+1$ кусок. При этом каждый кусок лежит ровно в одной «старой» части плоскости.



Тогда ясно, что каждый кусок прямой разбивает свою часть плоскости на две. То есть, делает вклад $+1$ к количеству частей. Значит, всего добавится $n+1$ часть.

$$\frac{n(n+1)}{2} + 1 + n + 1 = \frac{(n+1)(n+2)}{2} + 1$$

Переход индукции доказан. □

2.3 Задача 3

Шеренга новобранцев стоит перед старшиной. Он командует «налево». По неопытности часть солдат поворачивается налево, а часть – направо. После этого каждую секунду происходит следующее: солдаты, оказавшиеся друг к другу лицом, понимают, что произошла ошибка, и поворачиваются кругом. В следующую секунду ситуация повторяется. Докажи, что рано или поздно шеренга встанет неподвижно. Эту задачу можно было бы решить с помощью полуинвариантов. Но сейчас мы предлагаем решить ее индукцией по числу солдат.

Доказательство. Будем доказывать индукцией по n (числу солдат).

База индукции при $n = 1$ очевидна.

Рассмотрим $n+1$ солдата. Если к нему никто никогда не поворачивается лицом, то забываем про этого солдата и рассматриваем оставшихся n.

Если же к нему кто-нибудь когда-нибудь повернется лицом, то он отвернется от этого человека и будет смотреть в воздух, и уже больше точно никогда не повернется и опять перейдем к оставшимся n солдатам. □

2.4 Задача 4

Скорее всего, в задаче опечатка, иначе она из милой доброй задачи про рекурру превращается в жуткий гроб с бесконечным счётом. Буду решать в такой формулировке:

Реши рекуррентное уравнение:

$$x_0 = 43, x_1 = 70, \quad x_{n+2} = 29x_{n+1} - 28x_n$$

Решаем характеристическое уравнение.

$$\lambda^2 - 29\lambda + 28 = 0$$

Корни:

$$\lambda_1 = 28, \quad \lambda_2 = 1$$

Общее решение:

$$x_n = A \cdot 28^n + B$$

Найдем константы:

$$\begin{cases} A + B = 43, \\ 28A + B = 70 \end{cases} \implies A = 1, \quad B = 42$$

Итоговое решение:

$$x_n = 28^n + 42$$

Ответ: $x_n = A \cdot 28^n + B$

2.5 Задача 5

Зайчик стоит у подножия лесенки высотой n ступенек. За один прыжок он может подняться либо на одну, либо на две ступеньки вверх. Сколькими способами он может добраться до верха?

Пусть x_k – число способов прыгнуть в k-ую ступеньку. Ясно, что $x_k = x_{k-1} + x_{k-2}$, так как зайчик может прыгнуть в k-ую ступеньку либо с $k - 1$ ступеньки, либо с $k - 2$ ступеньки.

Тогда это просто задача про рекуррентное уравнение вида

$$x_k = x_{k-1} + x_{k-2}$$

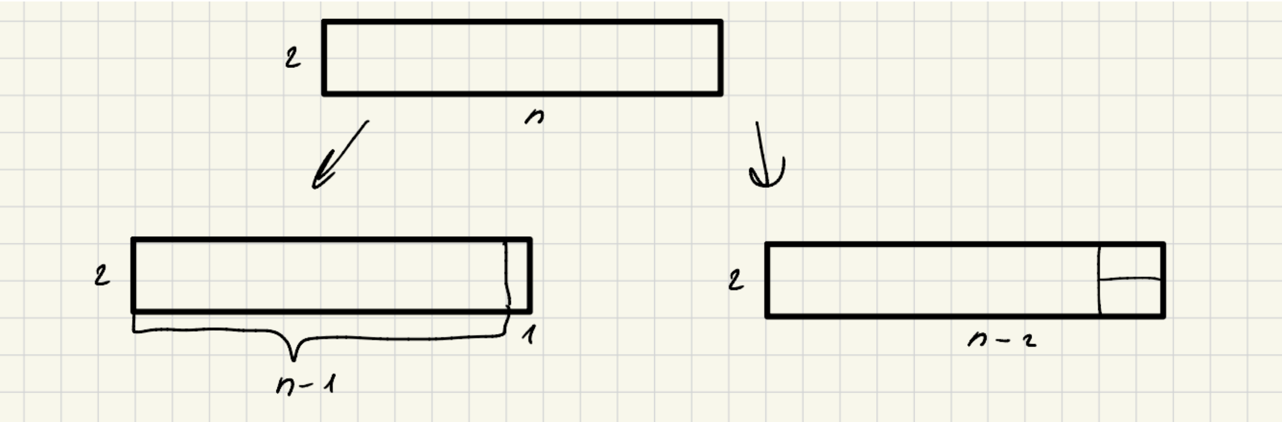
Ясно, что на первую ступеньку зайчик может прыгнуть только с самого низа (1 способ), а на вторую – либо с самого низа, либо с первой ступеньки (2 способа).

Тогда $x_n = F_{n+2}$, где F_i - i-ое число Фибоначчи.

Ответ: $x_n = F_{n-2}$

2.6 Задача 6

Есть 2 принципиально разных способа получить прямоугольник $2 \times n$.



Либо из прямоугольника $2\times(n-1)$ прибавлением одной доминошки, либо из прямоугольника $2\times(n-2)$ прибавлением двух доминошек.

Получаем рекуррентное уравнение $x_n = x_{n-1} + x_{n-2}$, где x_i – число способов получить прямоугольник $2 \times i$

Тогда ясно, что ответ на задачу – это числа Фибоначчи (возможно, с каким-то сдвигом).

$x_1 = 1$ (одна вертикальная доминошка)

$x_2 = 2$ (две вертикальных, либо две горизонтальных доминошек).

Тогда ясно, что $x_i = F_{i+2}$

Ответ: F_{i+2}

2.7 Задача 7

Докажи тождества для всех натуральных m, n:

а) **(тождество Кассини)** $F_{n+1} \cdot F_{n-1} - F_n^2 = (-1)^n$

б) $F_{m+n} = F_m \cdot F_{n+1} + F_{m-1} \cdot F_n$

в) $F_{2n+1} = F_n^2 + F_{n+1}^2$

2.7.1 Пункт Б

$$F_{m+n} = F_m \cdot F_{n+1} + F_{m-1} \cdot F_n$$

Доказательство. Проверим базу при $n = 1$:

$$F_{m+1} = F_m \cdot F_2 + F_{m-1} F_2 = F_m + F_{m-1}$$

Верность равенства очевидна в силу определения чисел Фибоначчи. Заметьте, что это равенство верно при любом m (ну то есть я никак не пользуюсь тем, чему равно m).

Предположим, что $F_{m+n} = F_m \cdot F_{n+1} + F_{m-1} \cdot F_n$ при всех значениях $\leq n$.

Докажем, что тогда формула верна и для $n + 1$, то есть покажем справедливость формулы:

$$F_{m+n+1} = F_m \cdot F_{n+2} + F_{m-1} \cdot F_{n+1}$$

По определению чисел Фибоначчи:

$$F_{m+(n+1)} = F_{m+n} + F_{m+(n-1)}$$

Воспользуемся предположением индукции:

$$F_{m+(n+1)} = F_m \cdot F_{n+1} + F_{m-1} \cdot F_n + F_m \cdot F_n + F_{m-1} \cdot F_{n-1} = F_m(F_{n+1} + F_n) + F_{m-1} \cdot (F_n + F_{n-1}) = F_m \cdot F_{n+2} + F_{m-1} \cdot F_{n+1}$$

Это доказывает переход по n. А для m не надо доказывать, так как мы нигде его не фиксировали. $\square$

2.7.2 Пункт В

$$F_{2n+1} = F_n^2 + F_{n+1}^2$$

Доказательство. Воспользуемся результатом из пункта Б для $m := n+1, n := n$ $(F_{m+n} = F_m \cdot F_{n+1} + F_{m-1} \cdot F_n)$ Получим:

$$F_{n+(n+1)} = F_{n+1} \cdot F_{n+1} + F_n \cdot F_n = F_n^2 + F_{n+1}^2$$

□

2.8 Задача 8

Ясно, что из условия возникает следующее рекуррентное уравнение:

$$x_n = x_{n-1} + 6x_{n-2}$$

Решаем характеристическое уравнение:
$\lambda^2 - \lambda - 6 = 0$
$\lambda_1 = -2, \lambda_2 = 3$
Тогда:

$$x_n = c_1 \cdot (-2)^n + c_2 \cdot 3^n \tag{1}$$

У нас есть два условия: $x_0 = 0, x_1 = 1$.
Подставим $n = 0$ и $n = 1$ в уравнение 1 и получим:

$$\begin{cases} 0 = c_1 + c_2 \\ 1 = -2c_1 + 3c_2 \end{cases}$$

Решая эту систему находим: $c_1 = -1/5, c_2 = 1/5$
Ответ: $x_n = \frac{1}{5}(3^n - (-2)^n)$

2.9 Задача 9

Пусть: A_n – количество способов добраться от A до A за n шагов.
B_n – количество способов добраться от A до B за n шагов.
C_n – количество способов добраться от A до C за n шагов.
D_n – количество способов добраться от A до D за n шагов.

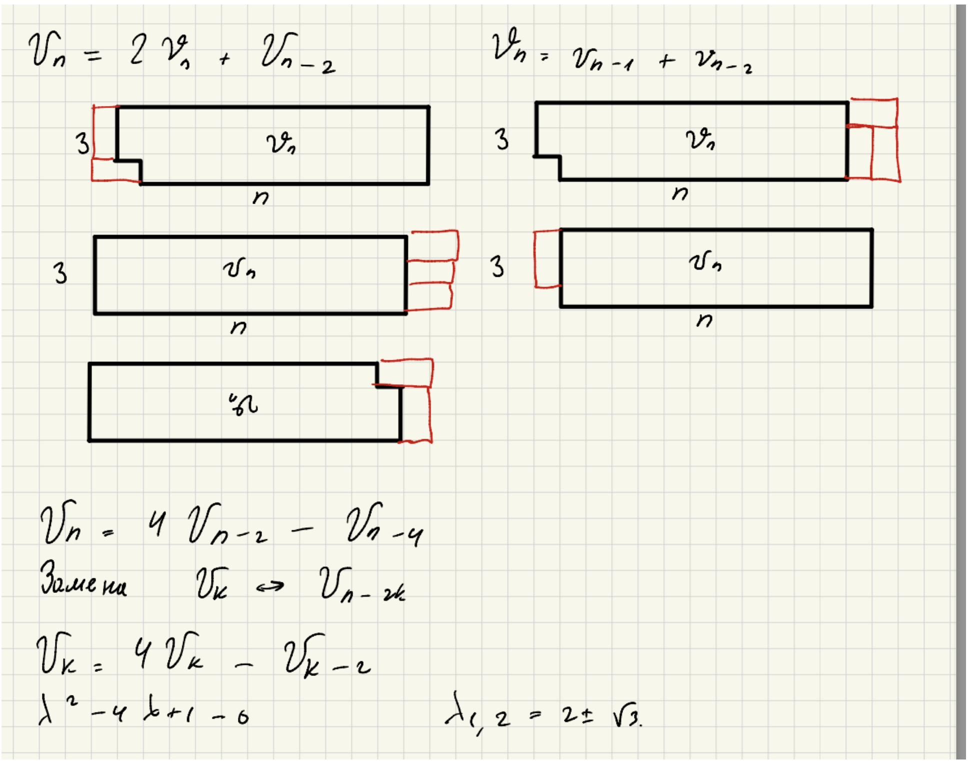
Заметим, что $A_n = C_n$. Действительно: в эти вершины я могу прийти только из B или D.
Аналогично: $B_n = D_n$
Так как в A можно попасть либо из B, либо из D, то: $A_n = 2B_{n-1}$
Аналогично: $B_n = 2A_{n-1}$. Подставим это в уравнение строчкой выше:

$$A_n = 4A_{n-2} \tag{2}$$

Ясно, что: $A_1 = 0,\ A_2 = 2$
Тогда решая рекуррентное уравнение 2, получаем: $A_{2k+1} = 0$
$A_{2k} = 2^{2k-1}$
Ответ: если $n = 2k+1$, то 0. если $n = 2k$, то 2^{2k-1}

2.10 Задача 10

Приложу картинку, из которой примерно можно понять идею решения, нет времени подробно расписывать.



3 Семинар 7

3.1 Задача 1

Есть 100 камней, выложенных в порядке возрастания весов. За какое наименьшее число взвешиваний на чашечных весах без гирь можно проверить или опровергнуть утверждение: «Любые пять камней вместе тяжелее любых трех»?

Решение. За одно. Достаточно взять 5 самых легких камней и 3 самых тяжелых.

3.2 Задача 2

Есть несколько мешков, в каждом из которых достаточно много монет. В одном мешке монеты фальшивые, а во всех других настоящие. Известен вес настоящей монеты и известно, что фальшивая монета на 1 грамм легче настоящей. Есть весы, которые показывают точный вес любого набора монет. Как за одно взвешивание определить мешок с фальшивыми монетами?

Решение. Берем из 1-го мешка одну монету, из 2-го мешка 2 монеты, $\ldots$, n-го мешка n монет,

Путь X – вес настоящей монеты. Тогда если бы все монеты были бы настоящими, то вес всех монет был бы $\dfrac{n(n+1)}{2} \cdot X$. Но на самом деле вес будет отличаться на l, так как в l-ом мешке лежат фальшивые монеты. Но тогда с помощью подсчёта разницы между ожидаемым и фактическим весом мы можем вычислить l.

3.3 Задача 3

Рассмотрим лексикографический порядок на всех сочетаниях из 10 по 5 элементов.

а) Какой номер будет иметь сочетание $\{2, 3, 5, 9, 10\}$?

б) Какое сочетание будет под номером 127?

Решение. Внутри сочетания числа не упорядоченны между собой, поэтому введение лексикографического порядка лишь упорядочивает их внутри сочетания.

3.3.1 Пункт Б

Сначала легче его решить его.

Пусть мы зафиксировали первое число единицей. Тогда для оставшихся 4 «мест» внутри сочетания остаётся 9 чисел. Тогда таких сочетаний (с единицей на первом месте) ровно $C_9^4 = \frac{9!}{4!5!} = 126$. То есть

126 сочетаний начинаются с единицы. Ясно, что 127 сочетание – это самое маленькое в смысле лексикографического порядка сочетание, которое начинается двойки. Поэтому ясно, что это будет сочетание $(2, 3, 4, 5, 6)$

Ответ: $(2, 3, 4, 5, 6)$

3.3.2 Пункт А

Ясно, что 2 и 3 – это минимальные первые числа для сочетаний, начинающихся с двойки.

Посчитаем число сочетаний, у которых потом идет 4. Ясно, что достаточно выбрать на 2 оставшихся места любые из 6 чисел. То есть существует $C_6^2 = 15$ способов.

Посчитаем число сочетаний, у которых потом идет 5. Ясно, что достаточно выбрать на 2 оставшихся места любые из 5 чисел. То есть существует $C_5^2 = 10$ способов.

Заметим, что 9 и 10 – это максимально возможные последние два числа в сочетании.

Поэтому ответ на задачу – $126 + 15 + 10 = 151$.

Ответ: 151

3.4 Задача 4

Илья хочет сообщить Владе некоторое слово из 10 букв, в котором четыре раза встречается буква A, три раза встречается буква B и три раза встречается буква C. Однако канал связи между ними позволяет передавать лишь нули и единицы. Ребята хотят заранее договориться о способе кодирования слов так, чтобы в худшем случае использовать минимально возможное число бит для передачи информации.

а) Разработайте такой способ кодирования. Какое количество бит потребуется для кодирования каждого слова при оптимальном способе кодирования?

б) Какой двоичный код будет соответствовать слову $BAABCCACAB$?

Решение. Посчитаем, сколько всего существует слов. Ясно, что нужно выбрать 4 места из 10 для А и из 6 оставшихся 3 места для В. На оставшиеся места встают С. Тогда всего слов $C_{10}^4 \cdot C_6^3 = 4200$.

Ясно, что слов длины k для алфавита только из двух символов ровно 2^k

Заметим, что $2^{12} = 4096$. То есть 12 бит не хватит, а вот 13 бит уже хватит, так как $2^{13} = 8192$.

3.4.1 Пункт А

Достаточно ввести лексикографический порядок на этих сочетаниях и отобразить номер каждого сочетания в двоичную систему.

3.4.2 Пункт Б

Нужно применить рассуждения, аналогичные третьей задаче. Не хочется расписывать это. Ответ на этот пункт, кстати, будет зависеть от того, как именно мы выберем наш лексикографический порядок. Так как буквы A, B, C можно по разному сравнить между собой и это даст разные порядки (и следовательно, разные ответы)

3.5 Задача 5

Фокусник решил показать зрителям фокус с угадыванием карты. В колоде у зрителя есть несколько карт, он загадывает одну из них. Фокусник раскладывает все карты на 4 стопки и узнает у зрителя, в какой стопке оказалась задуманная карта. При каком наибольшем количестве карт можно наверняка определить задуманную карту за 3 вопроса?

Ответ: 64

Пример: Ясно, что с 64 картами всё получится, нужно лишь делить каждый каждый раз стопку на 4. За 3 хода ровно и справимся.

Оценка: Закодируем ответы зрителя в четверичной системе. Например если он выбирал стопки в порядке 1, 3, 4, 2, то этому будет соответствовать число 0231_4. Таких последовательностей 64, поэтому если карт будет 65 и больше, то по принципу Дирихле какой-то последовательности будут соответствовать 2 карты.

3.6 Задача 6

Имеется 12 на вид неотличимых монет, из которых одна фальшивая. Все настоящие монеты весят одинаково, а фальшивая отличается от настоящих по весу, но неизвестно, в какую сторону (легче или тяжелее).

Имеются чашечные весы, на которых за одно взвешивание можно сравнить любые две группы монет по весу. За какое минимальное число взвешиваний гарантированно можно найти фальшивую монету и узнать, легче она или тяжелее, чем настоящие?

Ответ: 3

Оценка: Упростим нашу задачу, и допустим мы сразу знаем, тяжелее ли фальшивая монетка или легче. Даже в таком случае нам нужно минимум $\lceil \log_3 12 \rceil = 3$ взвешивания. Поэтому за два и меньше точно не получится.

Пример:

Сначала я опишу алгоритм, а потом объясню, почему он работает.

Алгоритм:

Нумеруем монеты в троичной системе вот так:

001	010	011	012
112	120	121	122
200	201	202	220

Теперь сначала я опишу, какие взвешивания я буду производить, а затем напишу, какую информацию от каждого взвешивания я буду запоминать.

Сначала я беру монетки с 1-ой цифрой 0 на левую чашу весов, с 1-ой цифрой 2 на правую чашу весов, с 1-ой цифрой 1 не кладу на весы.

Затем я беру монетки с 2-ой цифрой 0 на левую чашу весов, с 2-ой цифрой 2 на правую чашу весов, с 2-ой цифрой 1 не кладу на весы.

Затем я беру монетки с 3-ей цифрой 0 на левую чашу весов, с 3-ей цифрой 2 на правую чашу весов, с 3-ей цифрой 1 не кладу на весы.

После очередного взвешивания я приписываю себе на листочек по одной цифре по следующему правилу:

Если перевесила левая чаша весов, то я записываю себе на листочек цифру 0.

Если перевесила правая чаша весов, то я записываю себе на листочек цифру 2.

Если чаши весов равны, то я записываю на листочек цифру 1.

В результате я получу код из трёх цифр.

Если полученный код присутствует в списке занумерованных монет (например, 112), то монета с этим номером фальшивая и тяжелая.

Если полученного кода нет в списке, то фальшивая монета обязательно лёгкая. В полученном коде нужно заменить все нули на двойки, а все двойки на нули. Тогда мы обязательно получим номер фальшивой монеты.

Теперь объяснение, что это вообще был за алгоритм и почему он работает:

Смотрите, нумерация монет выбрана оооочень не случайно. Как минимум стоить заметить, что на каждой из трёх позиций каждая из цифр 0, 1 или 2 встречается ровно по 4 раза.

На каждом ходе мы имеем 3 кучки по 4 монеты. Причём на кучи мы разбиваем по цифрам на соответствующих разрядах. Значит на каждом шаге мы записываем цифру с тяжелых куч.

Если фальшивая монета была тяжелой, то мы выписали её код, так как мы выписывали цифры самых тяжелых куч. В частности, это объясняет, что для тяжелой монеты мы не можем получить код не из списка (и следовательно никаких замен нулей с двойками в этом случае нет и быть не может). Это лёгкий случай.

Если фальшивая монета была легкой, то этот случай сильно сложнее. Нужно объяснить: 1) почему мы получим код не из списка 2) почему замена нулей с двойками даст нужный код фальшивой монеты из списка.

Важное свойство нашей нумерации монет состоит в том, что при замене в любом номере нулей на двойки, а двоек на нули, мы обязательно получаем код, которого нет в списке.

Что происходит при взвешиваниях: если легкая монета не попадает на весы, то мы записываем единицу. Если легкая монета попадает на весы, то мы записываем ноль вместо двойки и двойку вместо нуля. То есть как раз получаем код не из списка. И при повторной такой замене мы получим нужный номер нашей легкой монеты.

3.7 Задача 7

Из 11 шаров два радиоактивны. Про любую кучку шаров за одну проверку можно узнать, имеется ли в ней хотя бы один радиоактивный шар (но нельзя узнать, сколько их). За какое наименьшее число проверок можно гарантированно выявить оба радиоактивных шара?

Ответ: 7

Пример: Разбиваем все шары на 5 групп по 2 и остаётся ещё 1 шар.

Проверяем для первых пяти групп, есть ли в каждой из групп радиоактивный элемент.

Если получилось 2 группы с радиоактивными элементами, то в каждой из групп достаточно проверить по одному элементу (ещё 2 проверки).

Если получилась 1 группа с радиоактивным элементом, в этой группе достаточно проверить один элемент, и взять ещё тот, который мы не проверяли (ещё 1 проверка).

Оценка: Оценки устроены в построении контр-примеров для каждого числа шаров в первом вопросе.

Покажу на примере не более двух шаров в первом вопросе.

Допустим, мы потратили вопрос, чтобы узнать, что в группе из двух шаров нет радиоактивного. Тогда у нас осталось 9 шаров и не более пяти вопросов.

Вариантов того, кто из них может быть радиоактивен ровно $C_9^2 = 36$. Комбинаций ответов «да» или «нет» на 5 вопросов ровно $2^5 = 32$.

$C_9^2 > 2^5$. Следовательно, вариантов для радиоактивности шаров больше, чем потенциальных комбинаций ответов.

Аналогично делается для любого другого числа шаров в первом вопросе. Это не очень идейно.

3.8 Задача 8

Есть 100-этажное здание. Известно, что если яйцо сбросить с высоты N-го этажа (или с большей высоты), то оно разобьется. Если его бросить с любого меньшего этажа, оно не разобьется. У нас есть всего два яйца, и если они оба разобьются, бросать больше будет нечего. За какое минимальное количество бросков наверняка получится определить N?

Ответ: 14

Дикий баян. Есть видео-разбор у Трушина. Вставлю текстовый разбор с первого сайта в гугле.

Обратите внимание, что независимо от того, с какого этажа мы бросаем яйцо №1, бросая яйцо №2, необходимого использовать линейный поиск (от самого низкого до самого высокого этажа) между этажом «повреждения» и следующим наивысшим этажом, при броске с которого яйцо останется целым. Например, если яйцо №1 остается целым при падении с 5-го по 10-й этаж, но разбивается при броске с 15-го этажа, то яйцо №2 придется (в худшем случае) сбрасывать с 11-го,12-го,13-го и 14-го этажей.

Предположим, что мы бросаем яйцо с 10-го этажа, потом с 20-го...

Если яйцо №1 разбилось на первом броске (этаж 10-й), то нам в худшем случае приходится проделать не более 10 бросков. Если яйцо №1 разбивается на последнем броске (100-й этаж), тогда у нас впереди в худшем случае 19 бросков (этажи 10-й, 20-й, ..., 90-й, 100-й, затем с 91-го до 99-го). Это хорошо, но давайте уделим внимание самому плохому случаю. Выполним балансировку нагрузки, чтобы выделить два наиболее вероятных случая.

В хорошо сбалансированной системе значение Drops(Egg1) + Drops(Egg2) будет постоянным, независимо от того, на каком этаже разбилось яйцо №1. Допустим, что за каждый бросок яйцо №1 «делает» один шаг (этаж), а яйцо №2 перемещается на один шаг меньше. Нужно каждый раз сокращать на единицу количество бросков, потенциально необходимых яйцу №2. Если яйцо №1 бросается сначала с 20-го, а потом с 30-го этажа, то яйцу №2 понадобится не более 9 бросков. Когда мы бросаем яйцо №1 в очередной раз, то должны снизить количество бросков яйца №2 до 8. Для этого достаточно бросить яйцо №1 с 39 этажа. Мы знаем, что яйцо №1 должно стартовать с этажа X, затем спуститься на X-1 этажей, затем — на X-2 этажей, пока не будет достигнуто число 100. Можно вывести формулу, описыващее наше решение: X + (X − 1) + (X − 2) + ... + 1 = 100 -> X = 14. Таким образом, мы сначала попадаем на 14-й этаж, затем на 27-й, затем 39-й. Так что 14 шагов — худший случай.

Как и в других задачах максимизации/минимазиции, ключом к решению является «балансировка худшего случая».

Разбор взят из книги Гейл Л. Макдауэлл «Cracking the Coding Interview» (есть в переводе).

3.9 Задача 9

Имеется n различных по весу монет. За одно взвешивание можно сравнить любые две монеты и узнать, какая из них тяжелее. Требуется найти вторую по весу монету среди данных n монет.

а) Докажите, что существует алгоритм, который гарантированно решает эту задачу за $n + log_2 n + O(1)$ сравнений.

б) Докажите, что более оптимального алгоритма не существует.

3.9.1 Пункт А

Алгоритм такой: выставляем все эти n монет в ряд, из пары 1-2 выбираем победителя, из пары 3-4 выбираем победителя, ..., из последней пары выбираем победителя. Если осталось число, то переносим его на следующий этап.

Делаем так до тех пор, пока не останется одно число. На это потребуется $n-1$ операция.

А затем проходимся по дереву тех, кто проигрывал у первой монетки. Там не более чем $\log_2 n$ монеток. Нужно из них выбрать самую тяжелую. Ясно, что на это потребуется $\log_2 n - 1$ операция.

3.9.2 Пункт Б

Доказательство. Если я знаю, что какая-то монетка является второй по весу, то уж я как минимум уверен, что она меньше по весу, чем первая монетка. Но тогда я точно знаю, какаю монетка является первой по весу (иначе как я был уверен в том, что мы меньше, чем первая по весу монетка?)

Значит, мне нужно как минимум $n-1$ сравнение потратить на то, чтобы выявить самую тяжелую монетку (доказательство есть в лонгриде/было на лекции).

Но тогда после $n-1$ сравнения мне нужно сравнить элементы, с которыми сравнивалась самая крупная монетка. Их уж точно не меньше, чем $\log_2 n - 1$, иначе у нас не было бы достаточно информации для вывода о том, что самая тяжелая монетка и в правду самая тяжелая.

Вот и получается оценка $n + \log_2 n + O(1)$ □

4 Семинар 7

4.1 Задача 1

а) Сколько существует различных графов на данных n вершинах?

б) Сколько существует различных ориентированных графов на данных n вершинах?

4.1.1 Пункт А

Решение. Способов выбрать неупорядоченную пару вершин

$$C_n^2 = \frac{n(n-1)}{2}$$

Для каждой неупорядоченной пары мы можем как проводить ребро, так и не проводить ребро. Тогда ясно, что графов на n вершинах всего $2^{C_n^2}$ штук.

4.1.2 Пункт Б

Решение. Способов выбрать упорядоченную пару вершин

$$n(n-1)$$

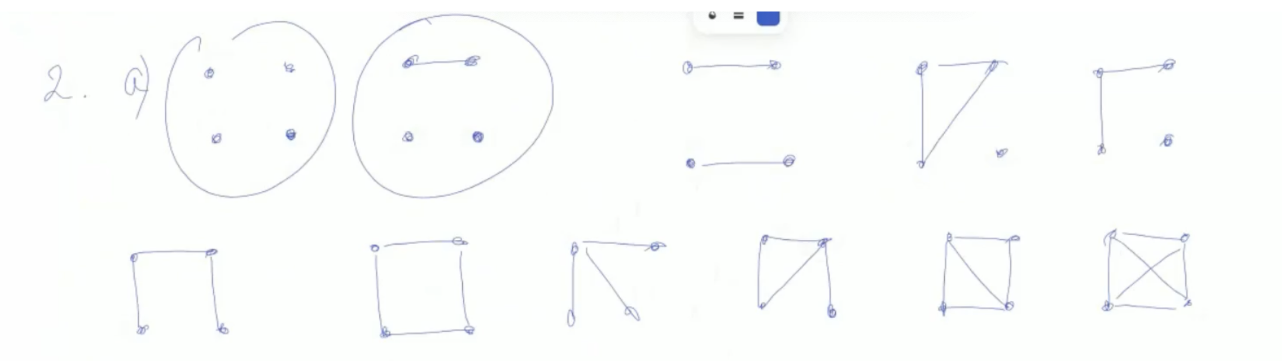
Для каждой упорядоченной пары мы можем как проводить ребро, так и не проводить ребро. Тогда ясно, что ориентированных графов на n вершинах всего $2^{n(n-1)}$ штук.

4.2 Задача 2

Сколько существует различных попарно неизоморфных графов: а) на 4 вершинах; б) на 5 вершинах?

Решение. Едва-ли тут можно придумать что-то сильно лучше, чем перебор картинок.

4.2.1 Пункт А



Ответ: 11

4.2.2 Пункт Б

Аналогично (гораздо больше картинок придется перебрать)[1]

Ответ: 34

4.3 Задача 3

Докажите, что в любом графе на $n \geq 2$ вершинах найдутся две вершины с одинаковой степенью.

Доказательство. Ясно, что возможные степени вершин в графе на n вершинах – это $0, 1, \ldots, n-1$.

При этом 0 и $n-1$ не могут присутствовать одновременно. То есть всего имеется $n-1$ доступный вариант степени для n вершин. Тогда по принципу Дирихле найдутся две вершины одинаковой степени □

4.4 Задача 4

а) Докажите, что если между двумя вершинами в графе есть путь, то между ними есть и простой путь.

б) Докажите, что если в графе есть цикл, то есть и простой цикл.

4.4.1 Пункт А

Доказательство. Неформально: выкидываем «ненужные» участки пути из пути и победа.

Чуть более формально: пусть у нас есть путь

$$v_0, e_1, v_1, \ldots e_k, v_k$$

Если в нём нет одинаковых вершин, то и доказывать нечего.

Если же в нём есть одинаковые вершины, то рассмотрим первое вхождение в эту вершину и последнее вхождение в эту вершину. Например, для вершины v_i:

$$v_0, e_1, v_1, \ldots, \quad \underbrace{v_i, e_{i+1}, \ldots,}_{\text{эту часть можно выкинуть}} \quad v_i, \ldots, e_k, v_k$$

Выкидываем из пути выделенную часть. Делаем так до тех пор, пока путь не станет простым. Ясно, что этот алгоритм остановится, так как граф на конечном числе вершин.

4.4.2 Пункт Б

Абсолютно аналогично пункту А. Нужно лишь заменить слово «путь» на слово «цикл». □

4.5 Задача 5

У Пети всего 28 одноклассников. У каждых двух из 28 различное число друзей в этом классе. Сколько друзей у Пети?[2]

Решение. Ясно, что в графе на 29 вершинах степень каждой вершины может быть одним из чисел $0, 1, \ldots, 28$.

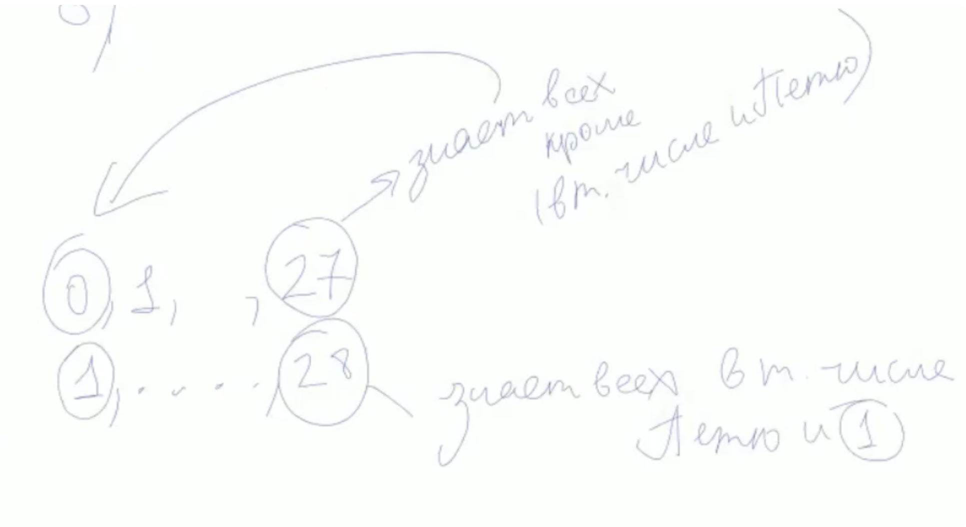
Предположим, что среди 28 одноклассников Пети есть степень вершины 0. В таком случае в графе нет степени вершины 28. Значит, из-за того, что у одноклассников попарно разное число друзей в классе, степени вершин у 28 одноклассников Пети в точности равны $0, 1, \ldots, 27$.

С другой стороны, если же степени вершины 0 среди 28 одноклассников Пети нет, то по безысходности у 28 одноклассников Пети степени вершин в точности равны $1, 2, \ldots, 28$.

Два случая разбираются абсолютно аналогично.

[1] Вообще вопрос о том, сколько существует неизоморфных графов на n вершинах, очевидно, изучен людьми менее кустарными методами. Результат для маленьких n выписан тут со ссылками на статьи

[2] На всякий случай уточню, что имеется в виду, что всего людей 29 (Петя и ещё 28 одноклассников)



Смотрите: в первом случае у нас есть человек со степенью вершины 0 и человек со степенью вершины 27. Ясно, что первый человек не дружит ни с кем. А второй дружит со всеми, кроме нуля. Ну значит, он дружит и с Петей.

Во втором случае у нас человек со степенью вершины 28. Ясно, что первый человек дружит только со вторым, а второй человек дружит вообще со всеми. Ну значит, он дружит и с Петей.

Теперь финт ушами. На этом шаге мы в любом из двух случаев поняли, что у Пети есть +1 друг.

Давайте отщипнем в обоих случаях первого и второго человека (то есть того, кто дружит с наименьшим числом людей и того, кто дружит с наибольшим числом людей). И будем рассматривать граф без них.

Тогда степень оставшихся людей уменьшиться ровно на 1. Действительно: они точно дружили со вторым (тот, который дружил с наибольшим числом людей) и точно не дружили с первым (тот, который дружил с наименьшим числом людей).

Последовательно будем применять к оставшимся графам подобное рассуждение. Тогда ясно, что у Пети 14 друзей.

Ответ: 14

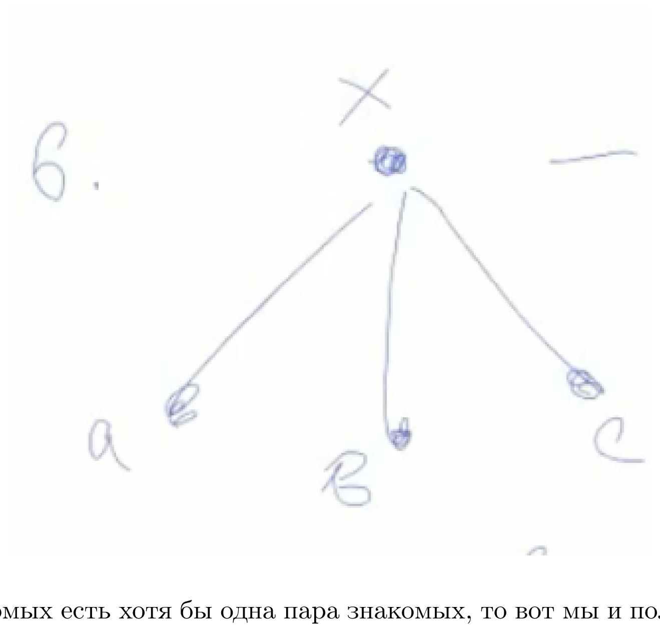
4.6 Задача 6

Докажите, что среди любых 6 людей найдутся или трое попарно знакомых, или трое попарно незнакомых.

Доказательство. Рассмотрим произвольного человека.

Ясно, что у него есть либо хотя бы 3 знакомых, либо хотя бы 3 незнакомых среди оставшихся.

Рассмотрим первый случай (второй разбирается симметрично).



Если среди 3 его знакомых есть хотя бы одна пара знакомых, то вот мы и получили тройку знакомых. А если нет – то мы получили тройку незнакомых. □