Определение и примеры

- Рекуррентное соотношение (от одной переменной) — уравнение, задающее функцию натурального аргумента n через ее значения при меньших n
  - функция натурального аргумента и последовательность — это одно и то же
- Условно-формальная запись: $f(0) = c$, $f(n) = F(n, f(0), \ldots, f(n-1))$
  - натуральный ряд удобно начинать с 0
  - недостаток записи: F имеет переменное число аргументов (это не функция)
  - чаще всего удается представить F как функцию

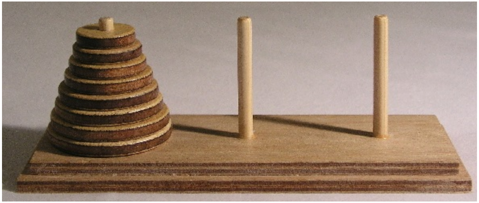
Примеры рекуррентных соотношений:

- $f(0) = b$, $f(n) = q \cdot f(n-1)$
  - геометрическая прогрессия со знаменателем q
- $f(0) = 1$, $f(n) = n \cdot f(n-1)$
  - $n!$
- $f(0) = 0$, $f(n) = f(n-1) + n^2$
  - $\sum_{k=1}^{n} k^2$
- $f(0) = 0$, $f(1) = 1$, $f(n) = f(n-1) + f(n-2)$
  - числа Фибоначчи
- $f(0) = 1$, $f(1) = 2$, $f(n) = f(n-1) + f(n-2)$
  - они же, но со сдвигом
- $f(0) = 1/2$, $f(n) = (1 + \sqrt{8}) f(n-1)(1 - f(n-1))$
  - частный случай логистической функции (https://en.wikipedia.org/wiki/Logistic_map)
- $f(0) = k$, $f(n) = \begin{cases} 3 \cdot f(n-1) + 1, & f(n-1) \text{ нечетно} \\ f(n-1)/2, & f(n-1) \text{ четно} \end{cases}$
  - утверждение о том, что $\forall k \in \mathbb{N} \exists n \in \mathbb{N} : f(n) = 1$ — известная открытая проблема («возможно, математика еще не готова к таким задачам» - Пал Эрдёш)

Можно выделить два основных типа задач на рекуррентные соотношения

1. Дана (неконструктивно) функция f, требуется научиться ее вычислять
   - пример: $f(n)$ есть число отношений эквивалентности на n-элементном множестве
   - ⋆ сюда же относится метод динамического программирования
2. Дана (рекуррентно) функция f, требуется найти замкнутую формулу для вычисления $f(n)$
   - ⋆ это называется решить рекуррентное соотношение
   - можно решить его, угадав формулу и доказав ее по индукции
   - ⋆ для некоторых классов рекуррентных соотношений существуют универсальные методы решения

Ханойская башня: постановка



Постановка задачи:

- имеются три стержня и n дисков, все диски разного диаметра
- в начальной конфигурации все диски образуют пирамиду на стержне 1
- в конечной конфигурации все диски образуют пирамиду на стержне 3
- ход состоит в перемещении одного диска с одного стержня на другой
    - остальные диски во время хода сдвигать нельзя
    - диск нельзя положить поверх диска меньшего размера
- найти минимальное число ходов $H(n)$, требуемое для перехода из начальной конфигурации в конечную

- Ход можно записывать как упорядоченную пару стержней
    - например, $2 \to 3$
- Очевидно, $H(0) = 0$, $H(1) = 1$
- $H(2) = 3$:
    - $1 \to 2$, $1 \to 3$, $2 \to 3$ — требуемая последовательность ходов
- Найдем рекуррентное соотношение для $H(n)$ $\Longrightarrow$

Ханойская башня: рекуррентное соотношение

- Пусть $M(n,i,j)$ — кратчайшая последовательность ходов, перемещающая n дисков со стержня i на стержень j

$\Rightarrow$ $H(n) = |M(n,1,3)|$
  - $|\;|$ обозначает длину последовательности
  - $|M(n,i,j)| = |M(n,1,3)|$ для любых i, j

★ Последовательность $M(n-1,1,2), 1 \to 3, M(n-1,2,3)$ переводит начальную конфигурацию дисков в конечную

$\Rightarrow$ $H(n) \leqslant 2H(n-1)+1$
  - $2H(n-1)+1$ ходов достаточно

- $M(n,1,3)$ включает ход, перемещающий самый большой (n-й) диск

⋆ n-й диск можно переместить только в момент, когда он единственный на своем стержне, а один из оставшихся стержней пуст

$\Rightarrow$ второй из оставшихся содержит пирамиду из $n-1$ диска

$\Rightarrow$ До первого перемещения n-го диска должно пройти не менее $H(n-1)$ шагов

- После последнего перемещения n-го диска — тоже не менее $H(n-1)$ шагов

$\Rightarrow$ $H(n) \geqslant 2H(n-1)+1$

★ Мы доказали рекуррентное соотношение $H(n) = 2H(n-1)+1$

Ханойская башня: замкнутая формула

Теорема

$H(n) = 2^n - 1$.

Доказательство:

- рекуррентное соотношение $H(n) = 2H(n-1) + 1$, $H(0) = 0$ определяет единственную функцию
- докажем по индукции, что это функция $2^n - 1$:
- база: $H(0) = 0 = 2^0 - 1$
- шаг: $H(n) = 2H(n-1) + 1 = 2 \cdot (2^{n-1} - 1) + 1 = 2^n - 1$ $\square$

Схема проведенного исследования функции $H(n)$:

неконструктивное определение

$\Rightarrow$ рекуррентное соотношение

$\Rightarrow$ замкнутая формула

Вычислим $E(n) = \int\limits_0^\infty x^n e^{-x} dx$, где $n \in \mathbb{N}$

$$E(0) = \int\limits_0^\infty e^{-x} dx = -e^{-x}|_0^\infty = 0 - (-1) = 1$$

К $E(n)$ применим интегрирование по частям:

$$E(n) = \int\limits_0^\infty x^n e^{-x} dx = \begin{bmatrix} u = x^n & du = nx^{n-1}dx \\ dv = e^{-x}dx & v = -e^{-x} \end{bmatrix} =$$

$$x^n(-e^{-x})|_0^\infty - \int\limits_0^\infty nx^{n-1}(-e^{-x})dx = 0 + n\int\limits_0^\infty x^{n-1}e^{-x}dx = nE(n-1)$$

★ $E(n)$ задается рекуррентным соотношением $E(n) = nE(n-1)$, $E(0) = 1$

$\Rightarrow$ $E(n) = n!$

★ Для вывода формулы Стирлинга $n! \sim \sqrt{2\pi n}\left(\frac{n}{e}\right)^n$ оценивают интеграл Эйлера, т.е. площадь под графиком функции $f_n(x) = x^n e^{-x}$

Интервальное расписание

Фрилансер зарабатывает деньги выполнением заказов и хочет максимизировать свой доход. Одна из возможных математических постановок —

Задача об интервальном расписании:

- даны k заказов, которые можно выполнить
- i-й заказ — это тройка (b_i, e_i, c_i), где $b_i, e_i, c_i \in \mathbb{N}$, $b_i \leqslant e_i$
- выполнение i-го заказа займет интервал времени $[b_i..e_i]$ и принесет доход c_i
- ⋆ какой максимальный доход можно получить, выполняя заказы, если в каждый момент времени можно выполнять не более одного заказа?

⋆ Метод динамического программирования позволяет найти и оптимальный список заказов, но мы ограничимся вычислением дохода

- Будем считать, что список заказов бесконечен и упорядочен по возрастанию «дедлайна» e_i
  - последовательности $\{b_i\}_1^\infty, \{e_i\}_1^\infty, \{c_i\}_1^\infty$ — параметры задачи
- Пусть $f(n)$ — максимальный доход с первых n заказов
  - положим $f(0) = 0$
- Через p_i обозначим наибольший номер k такой, что $e_k < b_i$
  - $p_i = 0$, если такого k не существует

⋆ если выполнен i-й заказ, то перед ним выполнен заказ с номером $\leqslant p_i$

★ $f(n) = \max\{f(n-1), f(p_n) + c_n\}$

- ⋆ можем вычислить $f(n)$ при любых допустимых значениях параметров

- Рекуррентное соотношение k-го порядка: $f(n) = F(n, f(n-1), \ldots, f(n-k))$
  - ⋆ здесь F — «нормальная» $(k+1)$-местная функция
  - ⋆ для задания функции таким соотношением нужно k начальных значений $f(0), \ldots, f(k-1)$

  Примеры:
  - числа Фибоначчи — соотношение второго порядка
  - факториал, логистическая функция, ханойская башня — первого порядка
  - ⋆ не являются соотношениями k-го порядка
  - соотношение из задачи про интервальные расписания (p_n может быть любым)
  - соотношения вида $f(n) = f(\lfloor n/2 \rfloor) + f(\lceil n/2 \rceil) + n$
    - ⋆ возникают при оценке сложности рекурсивных алгоритмов
- Рекуррентное соотношение $f(n) = F(n, f(n-1), \ldots, f(n-k))$ называется
  - ⋆ линейным, если $F = F(x_1, \ldots, x_k)$ — линейная функция с коэффициентами, зависящими от параметра n
    - т.е. $f(n) = a_1(n)f(n-1) + \cdots + a_k(n)f(n-k) + a(n)$
  - ⋆ линейным однородным, если $a(n) = 0$
  - ⋆ линейным с постоянными коэффициентами, если все $a_i(n)$ — константы
    - $a(n)$ может не быть константой

  Примеры:
  - логистическая функция $f(n) = r \cdot f(n-1)(1 - f(n-1))$: нелинейное
  - факториал $f(n) = n \cdot f(n-1)$:
    - линейное однородное с переменными коэффициентами
  - ханойская башня $f(n) = 2 \cdot f(n-1) + 1$:
    - линейное неоднородное с постоянными коэффициентами
  - числа Фибоначчи $f(n) = f(n-1) + f(n-2)$:
    - линейное однородное с постоянными коэффициентами

Решение линейных рекуррентных соотношений первого порядка

Запишем соотношение первого порядка в виде $f(n+1) = a(n)f(n) + b(n)$, $f(0) = a$

- при $a(n) = n+1$ и $b(n) = 0$ получается факториал

Теорема

$$f(n) = a \cdot \prod_{i=0}^{n-1} a(i) + \sum_{j=0}^{n-1} \left(b(j) \prod_{k=j+1}^{n-1} a(k) \right). \tag{1}$$

Доказательство:

Положим $g(n) = \frac{f(n)}{\prod_{i=0}^{n-1} a(i)}$, $g(0) = f(0) = a$

Запишем $f(n+1) - a(n)f(n) = b(n)$ и поделим обе части на $\prod_{i=0}^{n} a(i)$:

$$\frac{f(n+1) - a(n)f(n)}{\prod_{i=0}^{n} a(i)} = \frac{b(n)}{\prod_{i=0}^{n} a(i)} \quad \Rightarrow \quad \frac{f(n+1)}{\prod_{i=0}^{n} a(i)} - \frac{f(n)}{\prod_{i=0}^{n-1} a(i)} = \frac{b(n)}{\prod_{i=0}^{n} a(i)}$$

$$\Rightarrow \quad g(n+1) - g(n) = \frac{b(n)}{\prod_{i=0}^{n} a(i)}$$

Подставляя последнее равенство в $g(n) - g(0) = \sum_{j=0}^{n-1} (g(j+1) - g(j))$, получим

$g(n) = a + \sum_{j=0}^{n-1} \frac{b(j)}{\prod_{k=0}^{j} a(k)}$, откуда следует (1) $\square$

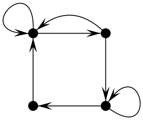
Линейное однородное рекуррентное соотношение с постоянными коэффициентами (ЛОРСПК) имеет вид

$$f(n) = a_1 f(n-1) + \cdots a_k f(n-k)$$

Если все коэффициенты $a_1, \ldots, a_k$ принадлежат кольцу $\mathbb{K}$, то $f : \mathbb{N} \to \mathbb{K}$

Примеры ЛОРСПК:

- $f(n) = q \cdot f(n-1)$
  - ⋆ геометрическая прогрессия со знаменателем q и первым членом $f(0)$
- $f(n) = f(n-1) + f(n-2)$
  - ⋆ числа Фибоначчи при $f(0) = 0$, $f(1) = 1$
- $f(n) = f(n-1) + f(n-2) + f(n-3)$
  - ⋆ числа Трибоначчи при $f(0) = 0$, $f(1) = 1$, $f(2) = 2$
- $f(n) = 2f(n-1) - f(n-3) + f(n-4)$
  - ⋆ при $f(0) = 4$, $f(1) = 7$, $f(2) = 13$, $f(3) = 24$ — число маршрутов длины n в графе



Постановка задачи

- Частным решением соотношения $f(n) = a_1 f(n-1) + \cdots + a_k f(n-k)$ называется любая функция f, удовлетворяющая этому соотношению
- Общим решением соотношения $f(n) = a_1 f(n-1) + \cdots + a_k f(n-k)$ называется множество всех его частных решений
  - ⋆ обычно под «решением» подразумеваем частное решение, т.е. функцию, а не множество
  - ⋆ любое ЛОРСПК имеет тривиальное решение $f(n) = 0$
  - ⋆ каждое частное решение определяется начальными значениями $f(j)$, $j = 0, \ldots, k-1$
  - ⋆ общее решение зависит от того, над каким кольцом/полем рассматривается соотношение
  - по умолчанию, мы рассматриваем соотношения над полем $\mathbb{R}$

Задача: дано ЛОРСПК $f(n) = a_1 f(n-1) + \cdots + a_k f(n-k)$, найти его общее решение

★ Общее решение ЛОРСПК является подмножеством линейного пространства $\mathbb{R}^\infty$ всех последовательностей действительных чисел

- с операциями сложения последовательностей и умножения последовательности на число

⋆ Пространство $\mathbb{R}^\infty$ бесконечномерно: его базисы счетны

- $\mathbb{R}^\infty = \langle \{\vec{e}_i\}_0^\infty \rangle$, где $\vec{e}_i = (0, \ldots, 0, 1, 0, \ldots, 0, \ldots)$ — последовательность, в которой на i-м месте стоит 1, а остальные элементы — нули

Лемма о подпространстве решений

Общее решение рекуррентного соотношения $f(n) = a_1 f(n-1) + \dots + a_k f(n-k)$ с коэффициентами из $\mathbb{R}$ является k-мерным подпространством в $\mathbb{R}^\infty$.

Доказательство: пусть S — общее решение

S — подпространство:

- ⋆ подмножество линейного пространства является подпространством тогда и только тогда, когда оно замкнуто относительно операций
- пусть $f_1(n), f_2(n)$ — решения, $\alpha \in \mathbb{R}$; тогда
  $f_1(n) + f_2(n) = a_1(f_1(n-1) + f_2(n-1)) + \dots + a_k(f_1(n-k) + f_2(n-k))$
  $f_1(n) + f_2(n)$ — решение
  $\alpha f_1(n) = a_1(\alpha f_1(n-1)) + \dots + a_k(\alpha f_1(n-k))$
  $\alpha f_1(n)$ — решение □

$dim(S) = k$:

- рассмотрим решения $e_i(n)$ с начальными условиями $e_i(j) = [i{=}j]$, $i,j = 0,\dots,k{-}1$
- ⋆ множество $\{e_i(n)\}_0^{k-1}$ линейно независимо, так как имеет ранг k:

$$
\begin{array}{rcl}
e_0 & = & (\,1, 0, \cdots 0,\ e_0(k),\ e_0(k{+}1),\ \cdots\,) \\
e_1 & = & (\,0, 1, \cdots 0,\ e_1(k),\ e_1(k{+}1),\ \cdots\,) \\
\vdots & & \quad \vdots\ \ \vdots\ \ \ddots\ \vdots \qquad \vdots \qquad\quad \vdots \qquad \vdots \\
e_{k-1} & = & (\,0, 0, \cdots 1,\ e_{k-1}(k),\ e_{k-1}(k{+}1),\ \cdots\,)
\end{array}
$$

- ⋆ $f(n) = f(0)e_0(n) + f(1)e_1(n) + \dots + f(k-1)e_{k-1}(n)$ для любого решения

$\Rightarrow$ $\{e_i(n)\}_0^{k-1}$ — базис $S \Rightarrow dim(S) = k$ □

Общее решение — первый подход

Лемма о подпространстве решений позволяет «решить» соотношение $f(n) = a_1 f(n-1) + \cdots + a_k f(n-k)$, записав общее решение в виде

$$f(n) = C_0 e_0(n) + \ldots + C_{k-1} e_{k-1}(n),\ C_0, \ldots, C_{k-1} \in \mathbb{R}$$

- ⋆ В чем дефект такого «решения»?
- ★ Оно не избавляет от рекурсии: функции $e_i(n)$ заданы тем же самым рекуррентным соотношением, что и функция $f(n)$
- Чтобы избавиться от рекурсии, нужно найти другой базис общего решения, состоящий из функций, значения которых можно вычислять нерекурсивно (например, экспоненциальных и полиномиальных функций)

Многочлен $\chi(x) = x^k - a_1x^{k-1} - \ldots - a_{k-1}x - a_k$ называется характеристическим многочленом рекуррентного соотношения $f(n) = a_1f(n-1) + \cdots + a_kf(n-k)$

Лемма о частных решениях

Пусть $\lambda \neq 0$. Функция $f(n) = \lambda^n$ является решением рекуррентного соотношения $f(n) = a_1f(n-1) + \cdots + a_kf(n-k)$ тогда и только тогда, когда λ — корень $\chi(x)$.

Доказательство:

- при $\lambda \neq 0$
  $\lambda^n = a_1\lambda^{n-1} + \cdots + a_k\lambda^{n-k} \Leftrightarrow \lambda^k = a_1\lambda^{k-1} + \cdots + a_k \Leftrightarrow \lambda$ — корень $\chi(x)$ $\square$

★ Неважно, над каким полем рассматривать рекуррентное соотношение: если его рассматривать над $\mathbb{C}$, комплексные корни характеристического многочлена также дадут решения

Лемма о независимости

Пусть $\lambda_1, \ldots, \lambda_s$ — различные корни характеристического многочлена рекуррентного соотношения $f(n) = a_1 f(n-1) + \cdots + a_k f(n-k)$. Тогда множество функций $\{\lambda_1^n, \ldots, \lambda_s^n\}$ линейно независимо.

Доказательство:

- выпишем функции $\lambda_1^n, \ldots, \lambda_s^n$:

$$\begin{array}{l} \lambda_1^n = (\ 1,\ \lambda_1,\ \cdots\ \lambda_1^{s-1},\ \lambda_1^s,\ \cdots\ ) \\ \lambda_2^n = (\ 1,\ \lambda_2,\ \cdots\ \lambda_2^{s-1},\ \lambda_2^s,\ \cdots\ ) \\ \quad\vdots \qquad\ \vdots \quad \vdots \quad \ddots \quad \vdots \qquad \vdots \quad\ \vdots \\ \lambda_s^n = (\ 1,\ \lambda_s,\ \cdots\ \lambda_s^{s-1},\ \lambda_s^s,\ \cdots\ ) \end{array}$$

⋆ в первых s столбцах видим матрицу Вандермонда, определитель которой равен 0 только если $\lambda_i = \lambda_j$ для некоторых $i \neq j$

$\Rightarrow$ в нашем случае определитель $\neq 0 \Rightarrow$ множество линейно независимо □

Теорема об общем решении (случай простых корней)

Пусть характеристический многочлен рекуррентного соотношения $f(n) = a_1 f(n-1) + \cdots + a_k f(n-k)$ имеет k различных корней $\lambda_1, \ldots, \lambda_k \in \mathbb{R}$. Тогда общее решение этого соотношения имеет вид $f(n) = C_1\lambda_1^n + \ldots + C_k\lambda_k^n$, где константы $C_1, \ldots, C_k$ пробегают множество $\mathbb{R}$.

Доказательство следует из лемм о подпространстве решений, о частном решении и о независимости □

- ★ Можно заменить в формулировке теоремы $\mathbb{R}$ на $\mathbb{C}$; получим общее решение того же самого ЛОРСПК, только рассматриваемого как соотношение над $\mathbb{C}$
- ★ Если нужно получить общее решение над $\mathbb{R}$, а среди k различных корней характеристического многочлена есть комплексные, нужно заметить, что
  - ⋆ комплексные корни многочленов над $\mathbb{R}$ попарно сопряжены
  - ⋆ если числа λ_1 и λ_2 сопряжены, то λ_1^n и λ_2^n сопряжены для любого n
  - ⋆ если λ_1 и λ_2 сопряжены, то комплексные функции λ_1^n и λ_2^n порождают в $\mathbb{C}^\infty$ то же самое подпространство, что и вещественные функции $\lambda_1^n + \lambda_2^n$ и $\imath(\lambda_1^n - \lambda_2^n)$
  - $\Rightarrow$ каждую пару комплексно сопряженных функций λ_1^n и λ_2^n заменим в базисе на $\lambda_1^n + \lambda_2^n$ и $\imath(\lambda_1^n - \lambda_2^n)$, получая базис из вещественных функций

Общее решение

Теорема об общем решении (для произвольных корней)

Пусть характеристический многочлен рекуррентного соотношения $f(n) = a_1 f(n-1) + \cdots + a_k f(n-k)$ имеет s различных корней $\lambda_1, \ldots, \lambda_s \in \mathbb{C}$ с кратностями $m_1, \ldots, m_s$ соответственно, $m_1 + \cdots + m_s = k$. Тогда общее решение этого соотношения над $\mathbb{C}$ имеет вид

$$f(n) = (C_1 + \ldots + C_{m_1} n^{m_1 - 1})\lambda_1^n + \ldots + (C_{m_1+\ldots+m_{s-1}+1} + \ldots + C_k n^{m_s - 1})\lambda_s^n,$$

где константы $C_1, \ldots, C_k$ пробегают множество $\mathbb{C}$.

Пример: если характеристический многочлен имеет корни $\lambda_1 = 3$ кратности 1, $\lambda_{2,3} = \pm 2\imath$ кратности 2, $\lambda_4 = 1$ кратности 3, то общее решение выглядит как

$f(n) = C_1 3^n + (C_2 + C_3 n)(2\imath)^n + (C_4 + C_5 n)(-2\imath)^n + (C_6 + C_7 n + C_8 n^2)$

★ Для перехода к общему решению над $\mathbb{R}$ при наличии комплексных корней надо воспользоваться их сопряженностью (см. предыдущий фрагмент)

вместо $(2\imath)^n$ и $(-2\imath)^n$ нужно взять вещественные функции

- $(2\imath)^n + (-2\imath)^n = [n \text{ четное}] \cdot (-1)^{\frac{n}{2}} \cdot 2^{n+1}$
- $\imath((2\imath)^n - (-2\imath)^n) = [n \text{ нечетное}] \cdot (-1)^{\frac{n+1}{2}} \cdot 2^{n+1}$

! не забывайте скобку Иверсона!

Вторая лемма о частных решениях

Пусть характеристический многочлен $\chi(x)$ рекуррентного соотношения $f(n) = a_1 f(n-1) + \cdots + a_k f(n-k)$ имеет корень λ кратности не менее $m+1$. Тогда функция $f(n) = n^m \lambda^n$ является решением данного соотношения.

Доказательство:

- При $m = 0$ доказано ранее (лемма о частных решениях); далее $m > 0$
  - ⋆ λ является корнем первых m производных многочлена $\chi(x)$
  - ⋆ умножение многочлена на x не меняет кратности его ненулевых корней
  - ⋆ если многочлены $p_1(x)$ и $p_2(x)$ имеют общий корень λ кратности m_1 и m_2 соответственно, то у $p_1(n) \pm p_2(n)$ есть корень λ кратности $\min\{m_1, m_2\}$

$\Rightarrow$ λ является корнем многочленов

1. $x^n - a_1 x^{n-1} - \ldots - a_k x^{n-k}$
2. $x^{n+1} - a_1 x^n - \ldots - a_k x^{n-k+1}$
3. $(n+1)x^n - a_1 n x^{n-1} - \ldots - a_k(n-k+1)x^{n-k}$ (производная многочлена 2)
4. $nx^n - a_1(n-1)x^{n-1} - \ldots - a_k(n-k)x^{n-k}$ (вычли 1 из 3)

⋆ λ обращает многочлен 4 в ноль $\Rightarrow$ $n\lambda^n$ — решение нашего соотношения

⋆ умножим многочлен 4 на x, возьмем производную и вычтем многочлен 4:

$$n^2 x^n - a_1(n-1)^2 x^{n-1} - \ldots - a_k(n-k)^2 x^{n-k}$$

$\Rightarrow$ $n^2\lambda^n$ — решение нашего соотношения

- Повторяя m раз, получаем решения $n\lambda^n, n^2\lambda^n, \ldots, n^m\lambda^n$

- Жорданова клетка — это матрица вида $J = \begin{bmatrix} \lambda & 1 & 0 & \dots & 0 \\ 0 & \lambda & 1 & \dots & 0 \\ 0 & 0 & \lambda & \dots & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \dots & 0 & \lambda \end{bmatrix}$
  - ⋆ $J[i,i] = \lambda$ для всех i и некоторого $\lambda \in \mathbb{C}$; $J[i,i{+}1] = 1$; $J[i,j] = 0$ иначе
- Жорданова матрица — это блочно-диагональная матрица $J = \begin{bmatrix} \boxed{J_1} & & & 0 \\ & \boxed{J_2} & & \\ & & \ddots & \\ 0 & & & \boxed{J_r} \end{bmatrix}$,

  где все матрицы J_i — жордановы клетки (возможно, разных размеров)

★ Теорема Жордана: для любой матрицы $A \in \mathbb{C}^{k \times k}$ существует такая обратимая матрица T, что матрица $J = TAT^{-1}$ — жорданова

★ Равенство $A = T^{-1}JT$ можно использовать для возведения A в степень:

$$A^n = (T^{-1}JT)^n = T^{-1}J^nT$$

⇒ Достаточно уметь возводить в степень жордановы матрицы

Лемма о степени жордановой клетки

Пусть J — жорданова клетка размера t с числом λ. Тогда $J^n[i,j] = \binom{n}{j-i}\lambda^{n+i-j}$. (Полагаем $\binom{n}{x} = 0$ при $x < 0$ и $x > n$.)

⋆ Лемма утверждает, что $J^n = \begin{bmatrix} \lambda^n & n\lambda^{n-1} & \binom{n}{2}\lambda^{n-2} & \dots & \binom{n}{t-1}\lambda^{n-t+1} \\ 0 & \lambda^n & n\lambda^{n-1} & \dots & \binom{n}{t-2}\lambda^{n-t+2} \\ 0 & 0 & \lambda^n & \dots & \binom{n}{t-3}\lambda^{n-t+3} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \dots & 0 & \lambda^n \end{bmatrix}$

Доказательство по индукции: база $(n = 1)$ очевидна; шаг индукции:

$$J^{n+1}[i,j] = \sum_{k=1}^{t} J^n[i,k] \cdot J[k,j] = J^n[i,j-1] + J^n[i,j] \cdot \lambda =$$

$$\binom{n}{j-1-i}\lambda^{n+i-j+1} + \binom{n}{j-i}\lambda^{n+i-j+1} = \binom{n+1}{j-i}\lambda^{n+1+i-j} \qquad \square$$

Следствие: Для жордановой матрицы выполняется $J^n = \begin{bmatrix} \boxed{J_1^n} & & & 0 \\ & \boxed{J_2^n} & & \\ & & \ddots & \\ 0 & & & \boxed{J_r^n} \end{bmatrix}$

Теорема об общем решении (для произвольных корней)

Пусть характеристический многочлен рекуррентного соотношения $f(n) = a_1 f(n-1) + \cdots + a_k f(n-k)$ имеет s различных корней $\lambda_1, \ldots, \lambda_s \in \mathbb{C}$ с кратностями $m_1, \ldots, m_s$ соответственно, $m_1 + \cdots + m_s = k$. Тогда общее решение этого соотношения над $\mathbb{C}$ имеет вид

$$f(n) = (C_1 + \ldots + C_{m_1} n^{m_1 - 1})\lambda_1^n + \ldots + (C_{m_1 + \ldots + m_{s-1} + 1} + \ldots + C_k n^{m_s - 1})\lambda_s^n,$$

где константы $C_1, \ldots, C_k$ пробегают множество $\mathbb{C}$.

- По второй лемме о частных решениях мы знаем k специальных решений
  - вида $n^j \lambda_i^n$, где $i = 1, \ldots, s$; $j = 0, \ldots, m_i - 1$
- Теорема утверждает, что эти решения образуют базис пространства решений
  - которое имеет размерность k
- Доказать линейную независимость специальных решений, как в случае простых корней, не получится

★ Чтобы доказать теорему, мы покажем методами линейной алгебры, что любое решение является линейной комбинацией специальных решений

Переход к системе линейных уравнений

Для компактности записи, пусть $f_n = f(n)$; запишем систему линейных уравнений

$$\begin{cases} f_n & = & a_1 f_{n-1} + \ldots + a_k f_{n-k} \\ f_{n-1} & = & f_{n-1} \\ \ldots & = & \ldots \\ f_{n-k+1} & = & f_{n-k+1} \end{cases}$$

в матричном виде:

$$\begin{bmatrix} f_n \\ f_{n-1} \\ f_{n-2} \\ \vdots \\ f_{n-k+1} \end{bmatrix} = \begin{bmatrix} a_1 & a_2 & \ldots & a_{k-1} & a_k \\ 1 & 0 & \ldots & 0 & 0 \\ 0 & 1 & \ldots & 0 & 0 \\ \vdots & \ddots & \ddots & \ddots & \vdots \\ 0 & 0 & \ldots & 1 & 0 \end{bmatrix} \begin{bmatrix} f_{n-1} \\ f_{n-2} \\ f_{n-3} \\ \vdots \\ f_{n-k} \end{bmatrix}$$

- Пусть $\vec{f_n} = (f_{n+k-1}, \ldots, f_n)^{\perp}$, A — матрица системы

$\Rightarrow$ $\vec{f_n} = A\vec{f}_{n-1}$ для любого $n \geqslant 1$

$\Rightarrow$ $\vec{f_n} = A^n \vec{f_0}$ ($\vec{f_0}$ — вектор начальных значений функции f)

★ Задача: найти выражение для последней компоненты вектора, являющегося произведением степени известной матрицы на известный вектор

- степени матрицы A вычисляются через жорданову матрицу $J = TAT^{-1}$

- $\star$ Собственные числа треугольных матриц — это их диагональные элементы
- $\Rightarrow$ собственные числа жордановой матрицы J — это числа λ ее клеток
- $\star$ A и J — подобны, т.е. являются матрицами одного и того же линейного оператора в разных базисах, связанных матрицей перехода T
- $\Rightarrow$ A и J имеют одно и то же мультимножество собственных чисел
- $\Rightarrow$ нужно найти собственные числа A (корни характеристического многочлена A)

Лемма о характеристических многочленах

$|\lambda E - A| = \chi(\lambda)$

Доказательство: разложим определитель по первой строке (красный множитель — знак слагаемого, синий — определитель подматрицы в столбцах $1, \ldots, i-1$)

$$|\lambda E - A| = \begin{vmatrix} \lambda - a_1 & -a_2 & \ldots & -a_{k-1} & -a_k \\ -1 & \lambda & \ldots & 0 & 0 \\ 0 & -1 & \ldots & 0 & 0 \\ \vdots & \ddots & \ddots & \ddots & \vdots \\ 0 & 0 & \ldots & -1 & \lambda \end{vmatrix} =$$

$$(\lambda - a_1)\lambda^{k-1} + \sum_{i=2}^{k} (-1)^{i+1}(-a_i)(-1)^{i-1}\lambda^{k-i} = \lambda^k - a_1\lambda^{k-1} - \cdots - a_{k-1}\lambda - a_k \quad \square$$

Собираем все вместе

- $A = \begin{bmatrix} a_1 & a_2 & \dots & a_{k-1} & a_k \\ 1 & 0 & \dots & 0 & 0 \\ 0 & 1 & \dots & 0 & 0 \\ \vdots & \ddots & \ddots & \ddots & \vdots \\ 0 & 0 & \dots & 1 & 0 \end{bmatrix}$, $\vec{f_n} = A^n \vec{f_0}$

⋆ $A^n = T^{-1} J^n T$, J — жорданова
  - ⋆ теорема Жордана

⋆ На диагонали матрицы J стоят корни $\chi(x)$ — числа $\lambda_1(m_1$ раз$), \dots, \lambda_s(m_s$ раз$)$
  - ⋆ лемма о характеристических многочленах + подобие A и J

⋆ Размер жордановой клетки в J с числом λ_i не превосходит m_i $(i = 1, \dots, s)$

⋆ Ненулевые элементы J^n являются произведениями полиномов на экспоненты:
  - ⋆ по лемме о степенях жордановой матрицы,
    $\binom{n}{j-i}\lambda^{n+i-j} = \frac{\lambda^{i-j}}{(j-i)!} n(n-1)\cdots(n+i-j+1)\lambda^n = p(n)\lambda^n$

⋆ Матрицы T и T^{-1}, как и вектор $\vec{f_0}$, не зависят от n

$\Rightarrow$ Элементы матрицы $T^{-1}J^nT = A^n$ и вектора $\vec{f_n} = A^n\vec{f_0}$ — линейные комбинации произведений вида $p(n)\lambda^n$

Пример $\Longrightarrow$

Пример: пусть $J = \begin{bmatrix} \lambda & 1 & 0 \\ 0 & \lambda & 0 \\ 0 & 0 & \mu \end{bmatrix}$, $T[i,j] = t_{ij}$, $T^{-1}[i,j] = \tau_{ij}$; тогда

$$J^n = \begin{bmatrix} \lambda^n & \frac{n}{\lambda}\lambda^n & 0 \\ 0 & \lambda^n & 0 \\ 0 & 0 & \mu^n \end{bmatrix}, \; J^n T = \begin{bmatrix} (t_{11} + \frac{t_{21}}{\lambda} n)\lambda^n & (t_{12} + \frac{t_{22}}{\lambda} n)\lambda^n & (t_{13} + \frac{t_{23}}{\lambda} n)\lambda^n \\ t_{21}\lambda^n & t_{22}\lambda^n & t_{23}\lambda^n \\ t_{31}\mu^n & t_{32}\mu^n & t_{33}\mu^n \end{bmatrix},$$

$$T^{-1}J^n T = \begin{bmatrix} (\tau_{11}t_{11} + \tau_{12}t_{21} + \frac{\tau_{11}t_{21}}{\lambda} n)\lambda^n + \tau_{13}t_{31}\mu^n & (\ldots) & (\ldots) \\ (\tau_{21}t_{21} + \tau_{22}t_{21} + \frac{\tau_{21}t_{21}}{\lambda} n)\lambda^n + \tau_{23}t_{32}\mu^n & (\ldots) & (\ldots) \\ (\tau_{31}t_{21} + \tau_{32}t_{21} + \frac{\tau_{31}t_{21}}{\lambda} n)\lambda^n + \tau_{33}t_{33}\mu^n & (\ldots) & (\ldots) \end{bmatrix}$$

⋆ Любая функция, удовлетворяющая соотношению $f(n) = a_1 f(n-1) + \cdots + a_k f(n-k)$, имеет вид $f(n) = p_1(n)\lambda_1^n + \cdots + p_s(n)\lambda_s^n$, где $p_i(n)$ — многочлен степени не выше $m_i - 1$, $i = 1, \ldots, s$

$\Rightarrow$ $f(n)$ является линейной комбинацией специальных решений $\lambda_1^n, \ldots, n^{m_1-1}\lambda_1^n, \ldots, \lambda_s^n, \ldots, n^{m_s-1}\lambda_s^n$,

что и требовалось доказать □

Фибоначчи

- ★ Решим рекуррентное соотношение $f(n) = f(n-1) + f(n-2)$, задающее, в частности, числа Фибоначчи
  - Характеристический многочлен: $\chi(x) = x^2 - x - 1$
    - корни $x_{1,2} = \frac{1 \pm \sqrt{5}}{2}$
    - ⋆ другими словами, $x_1 = \phi$, $x_2 = -1/\phi$, где $\phi = 1.618033\ldots$ — золотое сечение
  - Общее решение: $f(n) = C_1\left(\frac{1+\sqrt{5}}{2}\right)^n + C_2\left(\frac{1-\sqrt{5}}{2}\right)^n$
- ★ Найдем частное решение для чисел Фибоначчи:

$$\begin{cases} C_1 + C_2 = 0 & f(0) \\ \frac{1+\sqrt{5}}{2} C_1 + \frac{1-\sqrt{5}}{2} C_2 = 1 & f(1) \end{cases}$$

$$\Rightarrow\ C_1 = \frac{1}{\sqrt{5}},\ C_2 = -\frac{1}{\sqrt{5}}$$

$$\Rightarrow\ f(n) = \frac{1}{\sqrt{5}}\left(\frac{1+\sqrt{5}}{2}\right)^n - \frac{1}{\sqrt{5}}\left(\frac{1-\sqrt{5}}{2}\right)^n$$

- ★ Второе слагаемое в явной формуле для чисел Фибоначчи всегда меньше $1/2$ и быстро стремится к 0 с ростом n; значит, на практике n-е число Фибоначчи можно получить, вычислив первое слагаемое в действительных числах за $O(\log n)$ операций и округлив до ближайшего целого

Решим еще два рекуррентных соотношения:

★ $f(n) = 4f(n-1) - 4f(n-2)$, $f(0) = f(1) = 1$

- Характеристический многочлен: $\chi(x) = x^2 - 4x + 4$, корни $x_{1,2} = 2$
- Общее решение: $f(n) = (C_1 + C_2 n)2^n$
- Подставим начальные значения:

$$\begin{cases} C_1 = 1 & f(0) \\ 2C_1 + 2C_2 = 1 & f(1) \end{cases} \Rightarrow C_1 = 1, C_2 = -1/2 \Rightarrow f(n) = \left(1 - \tfrac{n}{2}\right)2^n$$

★ $f(n) = 4f(n-1) - 6f(n-2) + 4f(n-3)$, $f(0) = 0, f(1) = 1, f(2) = 3$

- Характеристический многочлен: $\chi(x) = x^3 - 4x^2 + 6x - 4$
  - $\chi(x) = (x-2)(x^2 - 2x + 2) = (x-2)((x-1)^2 + 1)$; $x_1 = 2$, $x_2 = 1 + \imath$, $x_3 = 1 - \imath$
- Общее решение: $f(n) = C_1 2^n + C_2(1+\imath)^n + C_3(1-\imath)^n$
  - ⋆ $(1 \pm \imath)^n = \left(\sqrt{2}\right)^n \left(\cos\frac{\pi n}{4} \pm \imath \cdot \sin\frac{\pi n}{4}\right)$
- Подставим начальные значения:

$$\begin{cases} C_1 + C_2 + C_3 = 0 & f(0) \\ 2C_1 + (1+\imath)C_2 + (1-\imath)C_3 = 1 & f(1) \\ 4C_1 + 2\imath C_2 - 2\imath C_3 = 3 & f(2) \end{cases} \Rightarrow C_1 = \tfrac{1}{2}, C_2 = \tfrac{-1-\imath}{4}, C_3 = \tfrac{-1+\imath}{4}$$

$\Rightarrow f(n) = \frac{1}{2}2^n + \left(\frac{-1-\imath}{4}\right)(1+\imath)^n + \left(\frac{-1+\imath}{4}\right)(1-\imath)^n =$! выведите сами !

$= 2^{n-1} + \left(\sqrt{2}\right)^{n-2}\left(\sin\frac{\pi n}{4} - \cos\frac{\pi n}{4}\right)$

Неоднородные ЛРС с постоянными коэффицентами

Что делать, если дано линейное неоднородное рекуррентное соотношение с постоянными коэффициентами?

$$f(n) = a_1 f(n-1) + \cdots a_k f(n-k) + h(n) \text{ для некоторой функции } h(n)$$

Возможная стратегия — перейти к однородному соотношению более высокой степени

Примеры:

- Пусть $f(n) = a_1 f(n-1) + \cdots a_k f(n-k) + c$, где c — константа
  - $f(n) - f(n-1) = a_1(f(n-1) - f(n-2)) + \cdots + a_k(f(n-k) - f(n-k-1))$
  - ⇒ $f(n) = (a_1+1)f(n-1) + (a_2 - a_1)f(n-2) + \cdots + (a_k - a_{k-1})f(n-k) - a_k f(n-k-1)$
  - ⋆ задали линейным однородным соотношением порядка $k+1$
- ⋆ Если $p(n)$ — многочлен степени m, то $(p(n) - p(n-1))$ имеет степень $m-1$
- Пусть $f(n) = a_1 f(n-1) + \cdots a_k f(n-k) + p(n)$, где степень $p(n)$ равна m
  - записав $f(n) - f(n-1)$, можно выразить $f(n)$ соотношением порядка $k+1$, в котором неоднородность является многочленом $p(n) - p(n-1)$ степени $m-1$
  - ⇒ повторив процедуру еще m раз, получим однородное соотношение порядка $m+k+1$
- Пусть $f(n) = a_1 f(n-1) + \cdots a_k f(n-k) + c \cdot d^n$, где c, d — константы
  - записав $f(n) - d \cdot f(n-1)$, можно выразить $f(n)$ однородным соотношением порядка $k+1$
- ! Перейдите к однородному соотношению от соотношения
  $f(n) = a_1 f(n-1) + \cdots a_k f(n-k) + \sin \pi n$

★ Существуют случаи совместной рекурсии, когда две функции (или больше) задаются при помощи рекуррентных соотношений, использующих предыдущие значения обеих функций

★ Подобную «систему» рекуррентных соотношений иногда можно решить, преобразовав в обычные рекуррентные соотношения

Пример: $\begin{cases} f(n) = 2g(n-1) + f(n-2), & f(0) = 1, f(1) = 0 \\ g(n) = f(n-1) + g(n-2), & g(0) = 0, g(1) = 1 \end{cases}$

$$f(n) = 2g(n-1) + f(n-2) = 3f(n-2) + 2g(n-3) = 4f(n-2) - f(n-4)$$
$$g(n) = f(n-1) + g(n-2) = 3g(n-2) + f(n-3) = 4g(n-2) - g(n-4)$$

- характеристический многочлен: $x^4 - 4x^2 + 1$; корни: $\pm\sqrt{2 \pm \sqrt{3}}$
- можно честно решать систему четырех уравнений, но мы упростим:
- $f(n) = 0$ при нечетных n (по индукции) $\Rightarrow$ найдем $\bar{f}(n) = f(2n)$
- имеем $\bar{f}(n) = 4\bar{f}(n-1) - \bar{f}(n-2)$, $\bar{f}(0) = 1$, $\bar{f}(1) = 3$, корни $2 \pm \sqrt{3}$

$$\Rightarrow \begin{cases} C_1 + C_2 = 1 & \bar{f}(0) = 1 \\ (2+\sqrt{3})C_1 + (2-\sqrt{3})C_2 = 3 & \bar{f}(1) = 3 \end{cases}$$

$$\Rightarrow C_1 = \tfrac{3+\sqrt{3}}{6},\ C_2 = \tfrac{3-\sqrt{3}}{6}$$

$$\Rightarrow f(n) = \begin{cases} 0, & n \text{ нечетно} \\ \left(\frac{3+\sqrt{3}}{6}\right)(2+\sqrt{3})^{n/2} + \left(\frac{3-\sqrt{3}}{6}\right)(2-\sqrt{3})^{n/2} & n \text{ четно} \end{cases}$$

! Найдите $g(n)$

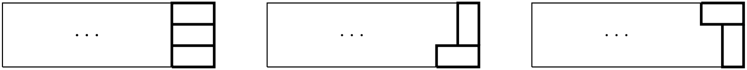
Совместная рекурсия (2)

Давайте решим комбинаторную задачу:

Задача: сколькими способами можно замостить прямоугольник размера $3 \times n$ костяшками домино размера 2×1? Костяшки разрешено поворачивать

- Пусть число способов равно $f(n)$
- $f(n) = 0$ при нечетных n, для четных составим рекуррентное соотношение

⋆ Все варианты замощения можно разбить на три группы по наличию или отсутствию костяшки, полностью лежащей в последнем столбце:



$\Rightarrow$ $f(n) = f(n-2) + 2g(n-1)$

- где $g(n)$ — число способов замостить прямоугольник $3 \times n$, оставив непокрытой одну угловую клетку справа

! Убедитесь, что $g(n) = f(n-1) + g(n-2)$

★ Функции $f(n)$ и $g(n)$ заданы системой рекуррентных соотношений

$$\begin{cases} f(n) = 2g(n-1) + f(n-2), & f(0) = 1, f(1) = 0 \\ g(n) = f(n-1) + g(n-2), & g(0) = 0, g(1) = 1 \end{cases}$$

... которую мы решили на предыдущем слайде

Нелинейные соотношения

При анализе алгоритмов часто возникают нелинейные рекуррентные соотношения

- Если мы ищем, есть ли число x в отсортированном массиве, мы применяем бинарный поиск
  - сравнили с элементом в середине массива и, в зависимости от результата, продолжили с нужной половиной
- Сложность бинарного поиска описывается соотношением $f(n) \leqslant 1 + f(\lceil n/2 \rceil)$
  - $\leqslant$ — потому что мы можем найти x при очередном сравнении

$\Rightarrow$ $f(n) \leqslant 1 + \lceil \log n \rceil$

- Рассмотрим более содержательный пример $\Longrightarrow$

Рассмотрим рекурсивный алгоритм сортировки слиянием (Mergesort):

Слияние отсортированных массивов:

Merge(array $B[1..k]$, $C[1..n]$)
  def array $D[1..n+k]$; $b, c, d \leftarrow 1$
  while $b \leqslant k$ and $c \leqslant n$
    if $B[b] < C[c]$
      $D[d] \leftarrow B[b]$; $b \leftarrow b+1$
    else
      $D[d] \leftarrow C[c]$; $c \leftarrow c+1$
    $d \leftarrow d+1$
  if b>k
    $D[c+k..n+k] \leftarrow C[c..n]$
  else
    $D[n+b..n+k] \leftarrow B[b..k]$
  return D

Сортировка:

Mergesort(array $A[1..n]$)
  if $n = 1$
    return A
  else
    $B \leftarrow$ Mergesort($A[1..\lceil n/2 \rceil]$)
    $C \leftarrow$ Mergesort($A[\lceil n/2 \rceil+1..n]$)
    return Merge(B, C)

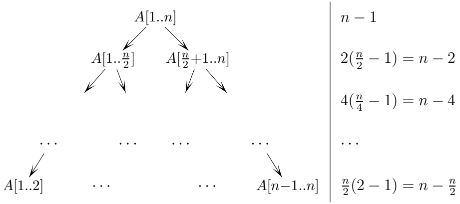
- ⋆ Mergesort относится к алгоритмам сортировки сравнением
  - ⋆ вычислительной сложностью такой сортировки считают количество сравнений символов, выполняемое при сортировке массива в худшем случае
  - ! докажите, что сложность функции Merge равна $n + k - 1$
- ⋆ массивы целых чисел можно сортировать, не сравнивая символы
  - ⋆ например, используя сортировку подсчетом (BucketSort)

Сложность сортировки слиянием (2)

Пусть $M(n)$ — сложность сортировки n-элементного массива слиянием

★ $M(n) \leqslant M(\lceil n/2 \rceil) + M(\lfloor n/2 \rfloor) + n - 1$
  - следует из определения MergeSort и замечания о сложности Merge

- Для оценки $M(n)$ решим соотношение $\bar{M}(n) = \bar{M}(\lceil n/2 \rceil) + \bar{M}(\lfloor n/2 \rfloor) + n - 1$
- Представим рекурсивные вызовы Mergesort в виде дерева:
  - на рисунке — «идеальный» случай $n = 2^k$
  - справа подсчитано число сравнений на каждом уровне дерева



- Число уровней равно $\log n$ (двоичный логарифм)

$\Rightarrow$ $\bar{M}(n) = n \log n - (n-1)$

★ В общем случае имеем $\lceil \log n \rceil$ уровней, с таким же подсчетом операций

$\Rightarrow$ $M(n) \leqslant n\lceil \log n \rceil - n + 1 \leqslant n\lfloor \log n \rfloor$ (детали восстановите самостоятельно)

⋆ Наиболее аккуратный подсчет дает $M(n) \leqslant n(\log n - \log e + \log\log e) + 1$

Нижняя оценка на сложность сортировок сравнением

★ Существует алгоритм (MergeSort), сортирующий любой массив длины n не более чем за $n(\log n - \log e + \log\log e) + 1$ сравнений символов

? какова нижняя оценка числа сравнений для любого алгоритма?

Теорема

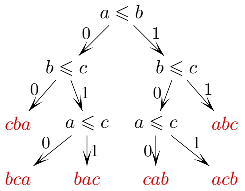
Любой алгоритм сортировки сравнением символов делает при сортировке массива длины n не менее $n \log n - n \log e + \frac{1}{2} \log n + O(1)$ сравнений в худшем случае

Доказательство:

- алгоритм выполняет сравнения во входном массиве A последовательно
- результат сравнения однозначно определяет следующее сравнение
- выполнив все сравнения, алгоритм знает сортирующую перестановку массива A

⇒ работу алгоритма можно представить деревом решения

Пример: дерево для алгоритма сортировки трехэлементного массива $[a, b, c]$



- дерево решения для сортировки — бинарное и имеет $\geqslant n!$ листьев

⇒ на самом длинном пути от корня до листа не менее $\log n!$ сравнений

- по формуле Стирлинга
  $\log n! = \log\left(\sqrt{2\pi n}\left(\frac{n}{e}\right)^n\right) + o(1) =$
  $n \log n - n \log e + \frac{1}{2} \log n + O(1)$ □

★ MergeSort делает не более $n \cdot \log\log e \sim n/2$ «лишних» сравнений

Мы встречали такие соотношения несколько раз:

- Биномиальные коэффициенты $\binom{n}{k} = \binom{n-1}{k-1} + \binom{n}{k-1}$
- Числа Стирлинга 1 рода $\left[{n+1 \atop k}\right] = n \cdot \left[{n \atop k}\right] + \left[{n \atop k-1}\right]$
- Числа Стирлинга 2 рода $\left\{{n+1 \atop k}\right\} = k \cdot \left\{{n \atop k}\right\} + \left\{{n \atop k-1}\right\}$
- Разбиения числа n на k слагаемых $P_k(n) = P_{k-1}(n-1) + P_k(n-k)$

⋆ Примеры, когда соотношение можно решить крайне редки
  - биномиальные коэффициенты

⋆ Обычно можно рассчитывать только на получение асимптотических приближений, верных для некоторых соотношений между переменными
  - $\left[{n+1 \atop k+1}\right] \sim \frac{n!}{k!}(\gamma + \ln n)^k$ при $k = o(\log n)$, где $\gamma \approx 0.577$ — константа Эйлера
  - $\left\{{n \atop k}\right\} \sim \frac{k^n}{k!}$ при фиксированном k
  - существует асимптотическая формула для $P_k(n)$ при $k = o(\sqrt{n})$
  - ...