

# Перечислительная комбинаторика и типы данных

Конспект: species, производящие функции, производные, маркеры, деревья, разбиения, симметрии и Burnside–Pólya

Этот текст реорганизует материал мини-курса из 12 листков: не по порядку упражнений, а по понятиям. Цель — иметь компактную карту определений, конструкций, примеров и тех скрытых соглашений, которые оказались особенно важны в ходе обсуждения.

*Версия для повторения и дальнейшей переработки*

## Содержание

|                                                         |    |
|---------------------------------------------------------|----|
| 1. Базовый язык: species и атомы                        | 3  |
| 1.1. Identity и value; 1.2. Сумма и произведение        | 3  |
| 2. OGF и EGF: что именно считается                      | 3  |
| 2.1. Автоморфизмы и вес формы в EGF                     | 4  |
| 3. Три базовых конструктора: SET, SEQ, CYC              | 4  |
| 3.1. Почему $G(0) = 0$                                  | 4  |
| 4. Композиция species                                   | 4  |
| 4.1. Bell polynomials; 4.2. Faà di Bruno                | 5  |
| 5. Маркеры, частные производные и факториальные моменты | 5  |
| 6. Производная species и pointing                       | 6  |
| 7. Рекурсивные типы и zipper                            | 6  |
| 8. Основные примеры                                     | 6  |
| 8.1. Списки                                             | 6  |
| 8.2. Полные плоские бинарные деревья (Каталан)          | 7  |
| 8.3. Корневые помеченные деревья                        | 7  |
| 8.4. Разбиения множества; 8.5. Перестановки             | 7  |
| 8.6. Endofunctions                                      | 8  |
| 9. Равенство рядов, явные биекции и естественность      | 8  |
| 10. Type algebra и «Seven Trees in One»                 | 8  |
| 11. Burnside, cycle index и Pólya                       | 8  |
| 11.1. SET; 11.2. Pólya substitution                     | 9  |
| 12. Семь неочевидных мест, которые стоит помнить        | 9  |
| 13. Мини-справочник формул                              | 10 |

## 1. Базовый язык: species и атомы

Комбинаторный вид (species)  $F$  ставит в соответствие каждому конечному множеству меток  $U$  множество  $F[U]$  структур на этих метках; переименование меток переносит структуры согласованным образом. Практически важнее всего помнить: species не выбирает элементы «с возвращением» — он организует данное множество атомов, используя каждый атом ровно один раз.

$X$  — выбранный нами атом размера 1. Символ  $1$  — единственная пустая структура размера 0. Поэтому смысл переменной зависит от того, что именно мы объявили атомом: у каталановского дерева  $X$  может означать лист; в другой модели  $X$  может означать каждую вершину.

$$X = \text{один атом выбранного сорта}; \quad 1 = \text{пустая структура}$$

**Ключевое соглашение №1.** «Размер» — это число  $X$ -атомов. «Забыть метки» не означает подставить  $X = 1$ :  $X$  продолжает учитывать размер.

### 1.1. Identity и value

Метка — это идентичность объекта, а не его значение. Два разных атома  $a$  и  $b$  могут иметь одинаковый payload (например, оба иметь значение `red`), оставаясь разными атомами. Поэтому повторяющиеся значения допустимы; недопустимо лишь дважды использовать один и тот же атом исходного множества.

**Ключевое соглашение №2.** В species различаются identity и value. Метки моделируют identity.

### 1.2. Сумма и произведение

Сумма  $F + G$  означает альтернативу: структура либо  $F$ -типа, либо  $G$ -типа. Произведение species существенно отличается от обычного декартова произведения на одном и том же наборе меток: сначала множество  $U$  разбивается на две непересекающиеся части  $A \sqcup B = U$ , затем строятся  $F[A]$  и  $G[B]$ .

$$(FG)[U] = \coprod_{U=A \sqcup B} F[A] \times G[B]$$

**Ключевое соглашение №3.** Произведение species = разбиение меток + декартово произведение структур на частях. Слово «независимы» здесь опасно, если забыть стадию распределения меток.

## 2. OGF и EGF: что именно считается

Обычная производящая функция (OGF) записывается как  $\sum a_n x^n$ . Экспоненциальная производящая функция (EGF) — как  $\sum a_n x^n / n!$ . Само по себе наличие OGF или EGF не является определением «непомеченного» и «помеченного» мира, но в наших задачах это естественные режимы счёта.

$$\text{OGF : } A(x) = \sum_{n \geq 0} a_n x^n \quad \text{EGF : } A(x) = \sum_{n \geq 0} a_n \frac{x^n}{n!}$$

В EGF число  $a_n$  — количество структур на фиксированном множестве из  $n$  различных меток. Деление на  $n!$  делает обычное произведение рядов совместимым с распределением меток между компонентами:

$$h_n = \sum_{k=0}^n \binom{n}{k} f_k g_{n-k}$$

потому что

$$\frac{x^k}{k!} \frac{x^{n-k}}{(n-k)!} = \binom{n}{k} \frac{x^n}{n!}$$

**Ключевое соглашение №4.** EGF «встроенно» кодирует распределение различных меток между компонентами.

### 2.1. Автоморфизмы и вес формы в EGF

Если  $A$  — непомеченная форма размера  $n$ , то число различных её маркировок равно  $n!/|\text{Aut}(A)|$ . После EGF-нормировки её вклад становится  $1/|\text{Aut}(A)|$ . Поэтому EGF можно читать и как сумму по непомеченным формам, взвешенным обратными размерами групп автоморфизмов.

$$\# \text{маркировок формы } A = \frac{n!}{|\text{Aut}(A)|}; \quad \text{EGF-вес формы} = \frac{1}{|\text{Aut}(A)|}$$

Пример: у корневой трёхвершинной звезды с двумя листьями  $|\text{Aut}| = 2$ ; она даёт 3 помеченные структуры и вклад  $3x^3/3! = x^3/2$ .

## 3. Три базовых конструктора: SET, SEQ, CYC

Пусть  $A$  — labeled species без структур размера 0. Тогда:

| Конструктор | Смысл                                           | EGF                  |
|-------------|-------------------------------------------------|----------------------|
| SET( $A$ )  | неупорядоченное множество $A$ -компонент        | $e^{A(x)}$           |
| SEQ( $A$ )  | упорядоченная последовательность $A$ -компонент | $\frac{1}{1 - A(x)}$ |
| CYC( $A$ )  | цикл $A$ -компонент                             | $-\log(1 - A(x))$    |

SET $_k$  можно мыслить как «SEQ $_k$ , у которого забыли порядок», то есть как факторизацию по перестановкам  $S_k$ . Это полезная рабочая картина, но нужно различать группу симметрий позиций и группу автоморфизмов конкретной структуры.

**Ключевое соглашение №5.** «SET = кортеж с забытым порядком» — полезная модель; но факторизация по  $S_k$  и стабилизатор конкретного объекта — связанные, но не тождественные понятия.

### 3.1. Почему $G(0) = 0$ при композиции и конструкторах компонентов

Если внутренняя компонента может быть пустой, можно добавлять сколько угодно пустых компонент, не меняя общий размер. Например, в SEQ( $1 + X$ ) объект размера 1 имеет бесконечно много представлений:  $[a]$ ,  $[, a]$ ,  $[a, ]$ , ... Поэтому для обычной композиции  $F \circ G$  требуется  $G(0) = 0$ .

Простой тест: каждая внутренняя  $G$ -компонента должна «съесть» хотя бы один исходный атом.

## 4. Композиция species

В  $F \circ G$  исходное множество меток сначала разбивается на непустые блоки. На каждом блоке строится  $G$ -структура. Затем сами блоки, рассматриваемые как объекты внешнего уровня, организуются в  $F$ -структуру.

$$(F \circ G)(x) = F(G(x)), \quad G(0) = 0$$

Важно: блоки становятся «атомами» только для внешней структуры. Размер всей конструкции всё равно равен числу исходных  $X$ -атомов. Если три метки разбились на один, два или три блока, все варианты по-прежнему лежат при  $x^3$ ; число блоков при необходимости маркируется отдельной переменной  $u$ .

**Ключевое соглашение №6.** Внешний уровень видит блоки как свои элементы, но  $x$  продолжает считать исходные атомы внутри блоков.

#### 4.1. Bell polynomials

Пусть  $G(x) = \sum_{j \geq 1} g_j x^j / j!$ . Если имеется  $m_j$  блоков размера  $j$ , то  $\sum j m_j = n$  и  $\sum m_j = k$ . Число способов получить такой тип блоков (с  $G$ -структурой на каждом блоке) равно

$$n! \prod_{j \geq 1} \left[ \frac{1}{m_j!} \left( \frac{g_j}{j!} \right)^{m_j} \right]$$

Сумма по всем допустимым наборам  $(m_1, m_2, \dots)$  даёт частичный Bell polynomial  $B_{n,k}(g_1, g_2, \dots)$ . Он считает внутренний слой композиции при фиксированном числе  $k$  блоков.

$$\frac{G(x)^k}{k!} = \sum_{n \geq k} B_{n,k}(g_1, g_2, \dots) \frac{x^n}{n!}$$

Если затем  $F(y) = \sum f_k y^k / k!$ , то

$$\left[ \frac{x^n}{n!} \right] F(G(x)) = \sum_k f_k B_{n,k}(g_1, g_2, \dots)$$

При  $g_j = 1$  Bell polynomials превращаются в числа Стирлинга второго рода:  $B_{n,k}(1, 1, \dots) = S(n, k)$ .

#### 4.2. Faà di Bruno как разбиение флажков

В производных композиции каждый выделенный атом должен попасть в какой-то внутренний  $G$ -блок. Поэтому слагаемые  $n$ -й производной соответствуют разбиениям множества из  $n$  различных флажков: блок размера  $j$  даёт  $G^{(j)}$ , а число блоков  $k$  даёт  $F^{(k)}(G)$ .

$$(F \circ G)'' = F'(G)G'' + F''(G)(G')^2$$

$$(F \circ G)''' = F'(G)G''' + 3F''(G)G'G'' + F'''(G)(G')^3$$

Коэффициент 3 во втором выражении — число разбиений трёх флажков типа  $2 + 1$ .

### 5. Маркеры, частные производные и факториальные моменты

Если  $u$  отмечает число компонент, то множитель  $u^k$  означает: объект имеет  $k$  таких компонент. Производная по  $u$  выбирает одну из них:

$$\frac{\partial}{\partial u}(u^k) = k u^{k-1}$$

То есть  $\partial_u$  выделяет одну компоненту и перестаёт учитывать её степень  $u$ ; оператор  $u \partial_u$  выделяет компоненту, но продолжает считать её в общем числе компонент.

$$u \frac{\partial}{\partial u} : \text{«выделить одну компоненту и продолжать её считать»}$$

Вторая производная даёт  $k(k-1)$ , то есть упорядоченную пару различных компонент (два различных флажка). Деление на  $2!$  превращает её в неупорядоченную пару. В общем случае  $u^r \partial^r / \partial u^r$  кодирует падающий факториал  $K(K-1) \cdots (K-r+1)$ .

Для перестановок  $P(x, u) = (1-x)^{-u}$ , где  $u$  считает циклы. При  $u = 1$  первая  $u$ -производная даёт общее число циклов во всех перестановках:  $n!H_n$ ; среднее число циклов равно  $H_n$ . Вторая факториальная производная ведёт к дисперсии  $H_n - H_n^{(2)}$ .

## 6. Производная species и pointing

Обычная производная species имеет жёсткое структурное определение:

$$F'[U] = F[U \sqcup \{*\}]$$

Здесь  $*$  — один специальный атом того же сорта, но он больше не учитывается переменной  $x$ . «Дырка» — удобная интерпретация этого специального атома, а конкретное представление производной может быть разным. Pointing возвращает обычный атом в дырку, сохраняя его выделенность:

$$XF' \longleftrightarrow \text{«}F\text{-структура с одним выделенным обычным атомом»}$$

На уровне EGF это  $xF'(x)$ .

**Ключевое соглашение №7.** Производная определена как  $F[U \sqcup \{*\}]$ ;  $L^2$ , zipper-контекст и другие формы — лишь естественные изоморфные представления этой структуры.

## 7. Рекурсивные типы и zipper

Если рекурсивный тип задан уравнением  $T = \Phi(X, T)$ , полезно различать обычную производную по атому  $X$  и производную конструктора по рекурсивному месту  $Y$ .

$$\Phi(X, Y) : Y = \text{место, куда вставляется целая рекурсивная структура}$$

$\Phi_Y$  описывает один frame — один родительский слой с дыркой вместо рекурсивного подде-рева. Последовательность frames от фокуса до корня образует zipper-контекст. В простых рекурсивных случаях:

$$\text{Context } C = \text{SEQ}(\Phi_Y(X, T))$$

Дифференцируя  $T = \Phi(X, T)$  по  $X$ , получаем

$$T' = \Phi_X + \Phi_Y T'$$

и структурно

$$T' = \text{SEQ}(\Phi_Y) \cdot \Phi_X$$

Первый множитель — путь/контекст до корня; второй — локальное содержимое непосредственно вокруг выбранного атома.

## 8. Основные примеры

### 8.1. Списки

$$L = 1 + XL = \text{SEQ}(X), \quad L(x) = \frac{1}{1-x}$$

Атомная производная: список с одной дыркой удобно кодируется двумя списками — слева и справа от дырки:

$$L' \cong L^2$$

Для рекурсивного конструктора  $\Phi(X, Y) = 1 + XY$  имеем  $\Phi_Y = X$ : один frame — это сохранённая голова списка. Полный рекурсивный контекст — последовательность таких голов.

## 8.2. Полные плоские бинарные деревья (Каталан)

Если атомы — листья:

$$B = X + B^2$$

OGF непомеченных форм имеет коэффициенты  $C_{n-1}$ . Если листья помечены, EGF формально совпадает с тем же рядом, но число labeled-деревьев на  $n$  фиксированных метках равно  $n!C_{n-1}$ . Для конструктора  $\Phi(X, Y) = X + Y^2$  один frame равен  $2Y$ : «пришли слева/справа + соседнее поддерево». Поэтому

$$B' \cong \text{SEQ}(2B) = \frac{1}{1 - 2B}$$

Здесь  $\Phi_X = 1$ , поэтому атомная производная совпадает с полным zipрег-контекстом. Это частный, но очень красивый случай.

## 8.3. Корневые помеченные деревья

$$T = X \cdot \text{SET}(T), \quad T(x) = xe^{T(x)}$$

По формуле Лагранжа число rooted labeled trees на  $n$  метках равно  $n^{n-1}$ . Для непомеченного корня  $U$  pointing даёт  $XU' = T$ , поэтому число обычных labeled trees равно  $n^{n-2}$  (формула Кэли). Связь

$$U = T - \frac{T^2}{2}$$

имеет топологический смысл:  $T$  — дерево с отмеченной вершиной,  $T^2/2$  — дерево с отмеченным неориентированным ребром; для каждого дерева  $V - E = 1$ .

## 8.4. Разбиения множества

$$\Pi = \text{SET}(\text{SET}_{\geq 1}(X)), \quad \Pi(x) = \exp(e^x - 1)$$

Обычная производная: блок с дыркой после удаления специального атома оставляет произвольное  $\text{SET}(X)$ , поэтому

$$\Pi' = \text{SET}(X) \cdot \Pi, \quad \Pi'(x) = e^x \Pi(x)$$

С маркером числа блоков:

$$\Pi(x, u) = \exp(u(e^x - 1)) = \sum_{n,k} S(n, k) u^k \frac{x^n}{n!}$$

Отсюда рекурсия  $S(n+1, k) = S(n, k-1) + kS(n, k)$  и PDE  $\Pi_x = u(\Pi_u + \Pi)$ .

## 8.5. Перестановки

$$\text{PERM} = \text{SET}(\text{CYC}(X)), \quad P(x) = \frac{1}{1-x}$$

С маркером числа циклов:

$$P(x, u) = (1-x)^{-u}$$

Это один из лучших примеров того, как marker-derivatives превращают комбинаторный класс в статистику.

## 8.6. Endofunctions

Функция  $f : S \rightarrow S$  распадается на связные компоненты; в каждой компоненте единственный ориентированный цикл, а к вершинам цикла подвешены rooted trees.

$$E = \text{SET}(\text{CYC}(T))$$

Поэтому

$$E(x) = \frac{1}{1 - T(x)}$$

и действительно коэффициенты дают  $n^n$ . Равенство EGF с  $\text{SEQ}(T)$  не означает изоморфизм species: одинаковый счёт слабее естественного структурного соответствия.

## 9. Равенство рядов, явные биекции и естественность

Важно различать три уровня утверждений:

- одинаковые числа объектов для каждого  $n$  (одинаковая EGF);
- явно построенная размер-сохраняющая биекция;
- естественный изоморфизм species, совместимый со всеми переименованиями меток.

В конечных размерах одинаковый счёт гарантирует существование некоторой абстрактной биекции, но не даёт канонической или естественной. Пример: endofunctions и  $\text{SEQ}(T)$  имеют одну EGF, но стандартная явная биекция использует выбранный порядок меток и потому не является естественной. Именно здесь species естественно соприкасаются с теорией категорий: species можно рассматривать как функторы  $\text{FinBij} \rightarrow \text{Set}$ , а естественный изоморфизм — как натуральное преобразование с биективными компонентами.

## 10. Type algebra и «Seven Trees in One»

Если забыть размер и оставить только рекурсивную форму полного бинарного дерева  $T \cong 1 + T^2$ , формальная кольцевая алгебра даёт  $T^2 = T - 1$ ,  $T^3 = -1$ ,  $T^6 = 1$ ,  $T^7 = T$ . Но вычитание не является положительной операцией над типами. Теорема Blass-Lawvere показывает, что несмотря на это существует настоящая положительная биекция  $T^7 \cong T$ . Этот пример полезен как предупреждение: формальная алгебра типов может подсказать истинную комбинаторную биекцию, но отрицательные коэффициенты требуют отдельного конструктивного содержания.

## 11. Burnside, cycle index и Pólya

Теорема Бернсайда считает орбиты действия группы:

$$\# \text{орбит} = \frac{1}{|G|} \sum_{g \in G} \# \text{Fix}(g)$$

Для labeled species на  $n$  метках действует  $S_n$ . Cycle index — это та же Burnside-сумма, но вместо немедленного забывания симметрии мы прикрепляем к каждой перестановке «паспорт» её циклического типа. Если  $c_j(\sigma)$  — число  $j$ -циклов, то паспорт равен  $p_1^{c_1} p_2^{c_2} \dots$ .

$$Z_{F,n} = \frac{1}{n!} \sum_{\sigma \in S_n} \text{Fix}_F(\sigma) \prod_{j \geq 1} p_j^{c_j(\sigma)}$$

Это ключевая мотивация cycle index: не новая загадочная generating function, а Burnside average с сохранённым циклическим типом каждой симметрии.

### 11.1. SET

Для SET любая перестановка фиксирует единственную SET-структуру, поэтому

$$Z_{\text{SET}} = \exp\left(p_1 + \frac{p_2}{2} + \frac{p_3}{3} + \dots\right)$$

Две фундаментальные подстановки:

$$\text{labeled EGF: } p_1 = x, p_j = 0 \ (j > 1) \implies e^x$$

$$\text{unlabeled OGF: } p_j = x^j \implies \frac{1}{1-x}$$

Вторая подстановка превращает каждый циклический моном в  $x^n$ , после чего Burnside усредняет фиксированные структуры и считает орбиты.

### 11.2. Pólya substitution

Для раскрасок цикл длины  $j$  должен быть одноцветным, если раскраска фиксируется данной перестановкой. При двух цветах  $r, b$  один  $j$ -цикл даёт вес  $r^j + b^j$ . Поэтому в cycle index подставляют

$$p_j \longmapsto r^j + b^j$$

Так cycle index превращает информацию о циклических типах симметрий в перечисление раскрасок с точностью до этих симметрий. Для более общего набора цветов вместо  $r^j + b^j$  подставляется соответствующая сумма  $j$ -х степеней весов цветов.

## 12. Семь неочевидных мест, которые стоит помнить

1.  $X$  — не «просто переменная», а выбранный атом размера; выбор атомизации меняет размер, GF и смысл производной.
2. Species организует фиксированное множество идентичностей и использует каждый атом ровно один раз; дубликаты значений — отдельная тема.
3. Произведение species сначала делит метки между факторами; обычное произведение EGF работает именно потому, что факториалы поглощают биномиальные коэффициенты.
4. В композиции блоки — элементы внешней структуры, но общий размер по-прежнему считается по исходным  $X$ -атомам.
5. SET удобно мыслить как кортеж с забытым порядком, однако факторизация по  $S_k$  и группа автоморфизмов конкретной формы — не одно и то же.
6. Производная species —  $F[U \sqcup \{*\}]$ ; «дырка»,  $L^2$ , zipper и т. п. — представления. Рекурсивная дырка  $\Phi_Y$  — другая операция.
7. OGF и EGF различаются прежде всего механизмом счёта: EGF нормирована так, чтобы правильно распределять различимые метки; вклад непомеченной формы равен  $1/|\text{Aut}|$ .

## 13. Мини-справочник формул

| Конструкция                  | Формула / смысл                                                          |
|------------------------------|--------------------------------------------------------------------------|
| $\text{SET}(A)$              | $\exp(A)$                                                                |
| $\text{SEQ}(A)$              | $\frac{1}{1-A}$                                                          |
| $\text{CYC}(A)$              | $-\log(1-A)$                                                             |
| Pointing                     | $XF' \longleftrightarrow xF'(x)$                                         |
| Composition                  | $(F \circ G)(x) = F(G(x)), \quad G(0) = 0$                               |
| Set partitions               | $\Pi = \text{SET}(\text{SET}_{\geq 1}(X)), \quad \Pi = \exp(e^x - 1)$    |
| Permutations                 | $\text{SET}(\text{CYC}(X)), \quad P = \frac{1}{1-x}$                     |
| Rooted labeled trees         | $T = X \cdot \text{SET}(T), \quad T = xe^T, \quad \# = n^{n-1}$          |
| Cayley trees                 | $XU' = T, \quad \# = n^{n-2}$                                            |
| Catalan leaves               | $B = X + B^2, \quad B' = \text{SEQ}(2B)$                                 |
| Endofunctions                | $E = \text{SET}(\text{CYC}(T)), \quad E = \frac{1}{1-T}, \quad \# = n^n$ |
| Cycle marker in permutations | $P(x, u) = (1-x)^{-u}$                                                   |
| Partition marker             | $\Pi(x, u) = \exp(u(e^x - 1))$                                           |
| Cycle index SET              | $\exp(p_1 + p_2/2 + p_3/3 + \dots)$                                      |

Главная общая картина: структуры  $\rightarrow$  действия переименований  $\rightarrow$  счёт (OGF/EGF)  $\rightarrow$  маркеры и производные  $\rightarrow$  композиция и разбиения  $\rightarrow$  симметрии (Burnside/cycle index/Pólya).