1 Перестановки

Напомним классическое определение. Перестановка — биективная функция, отображающая множество X на себя. Однако, в данной главе в качестве множества X мы будем рассматривать только множества целых чисел от 0 до $n-1$ (часто такое множество записывают как $\overline{0, n-1}$). Соответственно, определение становится таким:

Определение 1.1. Если $\pi : \overline{0, n-1} \to \overline{0, n-1}$ является биективной, то она — *перестановка.*

Обозначим множество всех перестановок множества $\overline{0, n-1}$ за S_n.

Традиционный способ задания перестановок — с помощью «двухэтажной» записи (обычно называемой *подстановкой*):

$$\begin{pmatrix} 0 & 1 & 2 & \dots & n-1 \\ \pi(0) & \pi(1) & \pi(2) & \dots & \pi(n-1) \end{pmatrix}$$

Однако, верхняя строчка в перестановках множества $\overline{0, n-1}$ всегда одинакова, поэтому мы будем её опускать, и записывать только нижнюю часть: $[\pi(0); \pi(1); \dots; \pi(n-1)]$. Такая запись не является канонической для математики, но зато она очень удобна для наших целей. Например, перестановка пяти элементов, меняющая местами 0 и 1 (и оставляющая на месте 2, 3 и 4), будет задаваться нами так: $[1; 0; 2; 3; 4]$

Также введём специальное обозначение для количества переставляемых элементов.

Определение 1.2. Пусть $\pi : \overline{0, n-1} \to \overline{0, n-1}$, *мощностью* перестановки π будем называть мощность множества отправления (прибытия). Будем обозначать эту мощность как $|\pi|$. Например, $|[0; 2; 1]| = 3$.

Также давайте введём понятие *префикса* перестановки:

Определение 1.3. Пусть $[a_0; a_1; \dots; a_{n-1}]$ — некоторая перестановка. Тогда последовательность чисел $[a_0; a_1; \dots; a_{k-1}]$ при $k \le n$ — *префикс* (начало) перестановки. В частности, префиксами являются пустая последовательность чисел $[]$ и вся исходная перестановка.

Легко заметить, что если последовательность чисел $[a_0; a_1; \dots; a_{k-1}]$ — последовательность попарно различных чисел из диапазона от 0 до $n-1$:

$$0 \le k \le n, \quad 0 \le a_i < n, \quad a_p \ne a_q \text{ при } q \ne q$$

то такая последовательность может быть продолжена до полной перестановки $(n-k)!$ способами.

1.1 Лексикографический порядок

Лексикографический порядок ещё называют словарным. Правила тут должны быть хорошо известны: чтобы сравнить два слова, нужно сравнить у них первые различающиеся буквы в одной и той же позиции. Например, слово *РУКА* идёт раньше в словаре, чем слово *РЫБА*, поскольку во второй позиции первого слова — буква *У*, а у второго слова — буква *Ы*, идущая позже.

Определение 1.4. Обозначим за $|S|$ длину слова S, и за S_i — символ строки номер i. Нумерацию по традиции будем вести с 0. Например, если $S =$ `абрикос`, то $|S| = 7$ и $S_4 =$ `к`.

Определение 1.5. Пусть даны два слова (S и T) одинаковой длины. Слово S меньше слова T (обозначим это отношение за $S \prec T$), если существует индекс i, что:

$$S_0 = T_0, \quad S_1 = T_1, \quad \dots, \quad S_{i-1} = T_{i-1}, \quad \text{но } S_i < T_i$$

Данное условие можно обобщить и на случай слов разной длины.

Определение 1.6. $S \prec T$, если выполнено одно из двух условий:

1. Либо существует индекс i, такой, что $0 \leq i < \min(|S|, |T|)$ и $S_i < T_i$, но для всех индексов j, $0 \leq j < i$, выполнено $S_j = T_j$,

2. либо $|S| < |T|$, и для всех индексов i $(0 \leq i < |S|)$, выполнено $S_i = T_i$

Мы легко могли бы написать функцию сравнения двух списков:

```
let rec is_less a b =
    match (a,b) with
        (a1::_,b1::_) when a1 < b1 -> true
      | (a1::ax,b1::bx) when a1 = b1 -> is_less ax bx
      | ([],_::_) -> true
      | _ -> false
```

Но в этом коде нет необходимости, поскольку операции сравнения определены и для списков, причём списки сравниваются лексикографически. Например, сравнения `[1;2;3] < [2;1;3]`, `[1] < [2;1]` и `[] < [9;7;10]` вернут `true`.

1.1.1 Упорядочение перестановок

Определим порядок на перестановках как лексикографический:

Определение 1.7. Перестановку π назовём меньшей перестановки σ (и обозначим это как $\pi \prec \sigma$, если найдётся такой индекс p, что:

1. $\pi(0) = \sigma(0), \quad \pi(1) < \sigma(1), \quad \ldots \quad, \pi(p-1) = \sigma(p-1)$

2. $\pi(p) < \sigma(p)$

В частности, $p = 0$ также можно рассматривать: $[0; 1] \prec [1; 0]$.

1.2 Несколько фактов о лексикографическом порядке на перестановках

Лемма 1.1. Рассмотрим перестановки равной мощности π и σ (пусть для определённости $\pi \preceq \sigma$), такие, что некоторый префикс этих перестановок совпадает:

$$\pi(0) = \sigma(0), \quad \pi(1) = \sigma(1), \quad \ldots \quad, \pi(p-1) = \sigma(p-1)$$

Тогда для того, чтобы перестановка τ находилась в лексикографическом порядке между π и σ $(\pi \preceq \tau \preceq \sigma)$, необходимо, чтобы и она имела тот же префикс:

$$\tau(0) = \pi(0), \quad \tau(1) = \pi(1), \quad \ldots, \quad \tau(p-1) = \pi(p-1)$$

Доказательство. Докажем от противного. Пусть это не так — тогда найдём минимальный отличающийся элемент перестановки — то есть, такой k, что $k < p$ и выполнены следующие равенства и неравенства:

1. $\tau(0) = \pi(0) = \sigma(0), \quad \tau(1) = \pi(1) = \sigma(1), \quad \ldots, \quad \tau(k-1) = \pi(k-1) = \sigma(k-1)$

2. $\tau(k) \neq \pi(k) = \sigma(k)$

Воспользовавшись определением сравнения перестановок, легко убедиться, что в этом случае перестановка τ либо меньше как π, так и σ (если $\tau(k) < \pi(k)$), либо больше (если $\tau(k) > \pi(k)$). В любом случае мы получаем противоречие с условием. $\square$

Лемма 1.2. Рассмотрим множество перестановок длины n с префиксом $[a_0; a_1; \ldots; a_{p-1}]$:

$$S_n^a = \{\pi \in S_n | \pi(0) = a_0 \,\&\, \pi(1) = a_1 \,\&\, \ldots \,\&\, \pi(p-1) = a_{p-1}\}$$

где a_i — какие-то заранее заданные значения.

Обозначим минимальную и максимальную перестановки в множестве за ρ и σ:

$$\rho = \min S_n^a, \quad \sigma = \max S_n^a$$

Тогда справедливо следующее:

$$\rho(p) < \rho(p+1) < \ldots < \rho(n-1)$$

$$\sigma(p) > \sigma(p+1) > \ldots > \sigma(n-1)$$

Иными словами, минимальная перестановка с данным префиксом — такая, в которой все оставшиеся элементы перестановки возрастают, а максимальная — в которой оставшиеся элементы убывают.

Например, $\min S_7^{[2;4;1]} = [2; 4; 1; 0; 3; 5; 6]$ и $\max S_7^{[2;4;1]} = [2; 4; 1; 6; 5; 3; 0]$.

Доказательство. В самом деле, пусть дана перестановка $\tau \in S_n^a$, покажем, что $\rho \preceq \tau$. Пусть это не так: то есть $\tau \prec \rho$, то есть найдётся минимальный такой t, что

$$\tau(0) = \rho(0), \quad \tau(1) = \rho(1), \quad \ldots, \quad \tau(t-1) = \rho(t-1), \quad \text{но } \tau t < \rho t$$

Заметим, что по определению $\tau \in S_n^a$ и $\rho \in S_n^a$, то есть перестановки τ и ρ совпадают в элементах с 0 по $p-1$. Значит, $t \geq p$.

Может ли $t = p$? Поскольку перестановки имеют одинаковые префиксы, оставшиеся (непрефиксные) множества элементов совпадают:

$$\{\rho(p), \rho(p+1), \ldots, \rho(n-1)\} = \overline{0, n-1} \setminus \{a_0, a_1, \ldots, a_{p-1}\} = \{\tau(p), \tau(p+1), \ldots, \tau(n-1)\}$$

Однако, $\rho(p) = \min\{\rho(p), \rho(p+1), \ldots, \rho(n-1)\} = \min\{\tau(p), \tau(p+1), \ldots, \tau(n-1)\}$, то есть $\rho(p) \leq \tau(p)$. По предположению же $\tau \prec \rho$, отсюда неизбежно $\rho(p) = \tau(p)$.

Осталось заметить, что в таком случае мы можем расширить префикс, добавив к нему общий для обеих перестановок элемент номер p:

$$\tau \in S_n^{[a_0; a_1; \ldots; a_{p-1}; \tau(p)]} = S_n^{[a_0; a_1; \ldots; a_{p-1}; \rho(p)]} \ni \rho$$

и, повторив рассуждение выше $n-p$ раз, мы покажем, что либо перестановки ρ и τ совпадают, либо в первом различии $\rho(t) < \tau(t)$ (что влечёт $\rho \prec \tau$).

Аналогично можно показать максимальность σ. $\square$

Лемма 1.3. Рассмотрим множество перестановок S_n и два префикса перестановок одинаковой длины: a и b ($|a| = |b| \leq n$), причём $a \prec b$. Тогда все перестановки из S_n^a меньше перестановок из S_n^b:

$$\forall \pi \in S_n^a \; \forall \tau \in S_n^b \; (\pi \prec \tau)$$

Доказательство. Очевидно из того, что сравнение перестановок идёт слева направо: сперва мы сравниваем префиксы, и приступаем к сравнению «послепрефиксной» части перестановок только если префиксы равны. А поскольку префикс a меньше префикса b ($a \prec b$), то и перестановки из S_n^a меньше перестановок из S_n^b. $\square$

1.3 Построение следующей перестановки по перестановке из S_n

Анализ вышеуказанных лемм позволяет нам предложить алгоритм построения следующей перестановки. Пусть дана перестановка $\pi = [a_0; a_1; \ldots; a_{n-1}]$. Тогда:

1. Рассмотрим такое минимальное число p, что

$$a_p > a_{p+1} > a_{p+2} > \ldots > a_{n-1}$$

   То есть, иными словами, убывающая последовательность имеет максимально возможную длину.

2. Среди чисел $a_p, a_{p+1} \ldots a_{n-1}$ найдём минимальное, бо́льшее a_{p-1} — пусть это a_t.

3. Тогда соберём искомую перестановку из трёх частей:

$$[a_0; a_1; \ldots; a_{p-2}] \mathbin{@} [a_t] \mathbin{@} \texttt{sort}(\{a_{p-1}, \ldots, a_{n-1}\} \setminus \{a_t\})$$

   То есть возьмём первые $p-1$ элементов исходной перестановки, дальше добавим a_t, и все оставшиеся элементы перестановки, отсортированные по возрастанию.

Теорема 1.1. Предложенный выше алгоритм действительно строит следующую перестановку в S_n, при условии, что $p > 0$.

Доказательство. Пусть мы по перестановке π получили перестановку ρ, пользуясь данным алгоритмом. По лемме 1.2 $\pi = \max S_n^{[a_0;a_1;\ldots;a_{p-2};a_{p-1}]}$ и $\rho = \min S_n^{[a_0;a_1;\ldots;a_{p-2};a_t]}$.

По построению, $[a_0; a_1; \ldots; a_{p-2}; a_{p-1}] \prec [a_0; a_1; \ldots; a_{p-2}; a_t]$, отсюда по лемме 1.3 все элементы из $S_n^{[a_0;a_1;\ldots;a_{p-1}]}$ меньше элементов из $S_n^{[a_0;a_1;\ldots;a_{p-2};a_t]}$, значит, $\pi \prec \rho$.

По лемме 1.1 между перестановками π и ρ нет перестановок с префиксом, отличным от $[a_0; a_1; \ldots; a_{p-2}]$. Каким же может быть $p-1$ значение в префиксе? Оно не может быть меньше a_{p-1} (иначе перестановка меньше π) и не может быть больше a_t (иначе перестановка больше ρ). А между a_{p-1} и a_t значений нет. При этом π — максимальная перестановка с префиксом $[a_0; a_1; \ldots; a_{p-2}; a_{p-1}]$, а ρ — минимальная с префиксом $[a_0; a_1; \ldots; a_{p-2}; a_t]$. Значит, π и ρ — соседние. $\square$