Дискретная математика

Григорьева Н.С.[1]

13.09.2023 - ...

[1]"Записал Сергей Киселев"

Оглавление

Глава 1

Комбинаторика

Лекция 1: Введение

13.09.2023

1.1 Основные определения

Определение 1. Перестановкой называется упорядоченный набор неповторяющихся элементов длины n, состоящий из элементов от 1 до n.

Число перестановок: $P_n = n!$

Определение 2. Размещением называется упорядоченный набор неповторяющихся элементов длины k, состоящий из элементов от 1 до n.

Число размещений: $A_n^k = n(n-1)(n-2) \cdot \ldots \cdot (n-k+1) =$
$= \frac{n(n-1)(n-2)\cdot\ldots\cdot(n-k+1)(n-k)!}{(n-k)!} = \frac{n!}{(n-k)!}$

Определение 3. Сочетанием называется набор неповторяющихся элементов длины k, состоящий из элементов от 1 до n.

Число сочетаний: $C_n^k = \frac{A_n^k}{k!} = \frac{n!}{k!(n-k)!}$

Определение 4. Перестановки с повторениями: $\overline{P_n} = \frac{n!}{n_1! \cdot n_2! \cdot \ldots \cdot n_k!}$

Определение 5. Размещения с повторениями: $\overline{A_n^k} = n^k$

Определение 6. Сочетания с повторениями: $\overline{C_n^k} = C_{n+k-1}^k$

Пример. (Толкование к сочетаниям с повторениями) Сколькими способами можно разложить пять одинаковых шаров по трём различным ящикам? На число шаров в ящике ограничений нет.

Решение:

Представим себе, что ящики стоят вплотную друг к другу. Три таких ящика — это фактически две перегородки между ними. Обозначим шар нулём, а перегородку — единицей. Тогда любому способу раскладывания пяти шаров по трём ящикам однозначно соответствует последовательность из пяти нулей и двух единиц; и наоборот, каждая такая последовательность однозначно определяет некоторый способ раскладывания. Например, 0010010 означает, что в первом ящике лежат два шара, во втором — два шара, в третьем — один шар; последовательность 0000011 соответствует случаю, когда все пять шаров лежат в первом ящике.

Теперь ясно, что способов разложить пять шаров по трём ящикам существует ровно столько же, сколько имеется последовательностей из пяти нулей и двух единиц. А число таких последовательностей равно C_7^2

1.2 Множества

Теорема 1. (Формула включений-исключений)

$|A_1 \cup A_2 \cup \ldots \cup A_n| = |\bigcup_{i=1}^n A_i| =$

$$= \sum_{i=1}^{n} |A_i| - \sum_{1 \le i < j \le n} |A_i \cap A_j| + \sum_{1 \le i < j < k \le n} |A_i \cap A_j \cap A_k| - \ldots + (-1)^{n-1} |A_1 \cap A_2 \cap \ldots \cap A_n|$$

Доказательство. (докажем по индукции)

1. База индукции: $n = 2 : |A_1 \cup A_2| = |A_1| + |A_2| - |A_1 \cap A_2|$
2. Переход индукции: $n \to n+1 :$

   $|A_1 \cup A_2 \cup \ldots \cup A_{n+1}| = |A_1 \cup A_2 \cup \ldots \cup A_n| + |A_{n+1}| - |(A_1 \cup A_2 \cup \ldots \cup A_n) \cap A_{n+1}| =$

   $$= \sum_{i=1}^{n} |A_i| - \sum_{1 \le i < j \le n} |A_i \cap A_j| + \sum_{1 \le i < j < k \le n} |A_i \cap A_j \cap A_k| -$$

   $- \ldots + (-1)^{n-1} |A_1 \cap A_2 \cap \ldots \cap A_n| + |A_{n+1}| - |(A_1 \cap A_{n+1}) \cup (A_2 \cap A_{n+1}) \ldots \cup (A_n \cap A_{n+1})| =$

   $$= \sum_{i=1}^{n} |A_i| - \sum_{1 \le i < j \le n} |A_i \cap A_j| + \sum_{1 \le i < j < k \le n} |A_i \cap A_j \cap A_k| -$$

   $$- \ldots + (-1)^{n-1} |A_1 \cap A_2 \cap \ldots \cap A_n| - (\sum_{i=1}^{n} |A_i \cap A_{n+1}| -$$

$$- \sum_{1 \le i < j \le n} |A_i \cap A_j \cap A_{n+1}| + \ldots (-1)^{n-1} |A_1 \cap A_2 \cap \ldots \cap A_{n+1}|) =$$

$$= \sum_{i=1}^{n+1} |A_i| - \sum_{1 \le i < j \le n+1} |A_i \cap A_j| + \sum_{1 \le i < j < k \le n+1} |A_i \cap A_j \cap A_k| -$$

$$\ldots + (-1)^n |A_1 \cap A_2 \cap \ldots \cap A_{n+1}|$$

□

1.3 Разбиения

Определение 7. Пусть A — множество. Имеется $A_1, A_2, \ldots, A_n$.
Совокупность этих множеств — разбиение, если: $A = \bigcup_{i=1}^{n} A_i; A_i \cap A_j = \varnothing$

Определение 8. Пусть у A есть разбиения $\mathcal{A}$ и $\mathcal{B}$
Тогда $\mathcal{B}$ — измельчение $\mathcal{A}$, если $\forall B_i \in \mathcal{B}\ \exists! A_j \in \mathcal{A} : B_i \subset A_j$

Определение 9. Произведение разбиений — разбиение, которое является измельчением $\mathcal{A}$ и $\mathcal{B}$ и является самым крупным измельчением.

Лекция 2: Разбиения, прямое произведение, нумерация

20.09.2023

Теорема 2. Произведение разбиений существует.

Доказательство. $\mathcal{A}, \mathcal{B}$ — разбиения.
Возьмем все множества вида $C_{ij} = A_i \cap B_j$

- C — измельчение $\mathcal{A}$, так как $\forall C_{ij}\ \exists A_i : C_{ij} \subset A_i$
- аналогично C — измельчение $\mathcal{B}$

Предположим, что F — измельчение, большее C, тогда:
$\forall F_k : \begin{cases} \exists A_i : F_k \subset A_i \\ \exists B_i : F_k \subset B_i \end{cases} \Rightarrow F_k \subset A_i \cap B_j \Rightarrow F_k \subset C_{ij}$ — C наибольшее из измельчений. □

Глава 2

Алгоритмы перебора

2.1 Перебор 0-1 векторов

Будем рассматривать множество B^m всех наборов из m битов, каждый из которых может быть нулем и единицей. Элемент множества B^m — вектор $(0, 0, 1, \dots, 1)$ длиной m. Количество элементов в множестве (мощность): $|B^m| = 2^m$.

Для того, чтобы создать вычислительный процесс, при котором на каждом шаге будет формироваться новый, не встречавшийся ранее, элемент рассматриваемого множества, достаточно заметить, что существует взаимнооднозначное соответствие между числами из $0 \dots 2^m - 1$ и наборами 0-1 векторов. Т. е. достаточно первым взять число 0 и его двоичное представление $(0, \dots, 0)$, а затем просто добавлять по единице, имитируя это на текущем наборе, пока мы не дойдем до набора из одних единиц.

Кроме рассмотренного способа перебора наборов, можно предложить другой алгоритм, который на каждом шаге меняет значение только одной компоненты:

> **Алгоритм.** (Перебор и нумерация 0-1 векторов в порядке минимального изменения)
>
> - создаем 2 набора x и y, каждый из m битов. Первоначально $x = y = (0, 0, 0, 0)$
> - прибавляем к x единицу и фиксируем позицию j, где произошло изменение.
> - изменить j-ую компоненту в наборе $y : y_j = 1 - y_j$
> - вернуть y

> **Пример.** (Рассмотрим на примере $m = 4$)

x	y	j
0000	0000	-
0001	0001	4
0010	0011	3
0011	0010	4
0100	0110	2
0101	0111	4

2.2 Перебор прямого произведения

Рассматриваем множество $M(1:k) = M_1 \times M_2 \times \ldots \times M_k$. Число элементов: $\prod_{i \in 1:k} m_i$, где $m_i = |M_i|$.

Будем считать, что каждое M_i состоит из m_i элементов, которые мы будем нумеровать от 0 до $m_i - 1$. Тогда каждый элемент $M(1:k)$ — последовательность неотрицательных чисел $(r_1, \ldots, r_k), r_i < m_i$

Общая формула перехода от элемента $(r_1, \ldots, r_k)$ к номеру этого элемента:

$$num(r_1, \ldots, r_k) = \sum_{i=1}^{k} \times (\prod_{j=1}^{i-1} m_j)$$

2.3 Перебор перестановок

Рассмотрим множество $T_k = M_1 \times M_2 \times \ldots \times M_k, M_i = \{0, 1, \ldots, i-1\}, |T_k| = k!$. Обозначим множество всех перестановок из k элементов через P_k.

Построим взаимнооднозначное соответствие между T_k и P_k. Возьмем перестановку $(r_1, \ldots, r_k)$ и сопоставим ей элемент $(t_1, \ldots, t_k)$ следующим образом: $\forall i \in 1:k$ найдем число значений, меньших r_i среди $r_{i+1}, \ldots, r_k$ — это число перепишем в качестве t_i.

Пример.

i	1	2	3	4	5	6	7	8
r_i	4	8	1	5	7	2	3	6
t_i	3	6	0	2	3	0	0	0

Чтобы получить перестановку по записи $(t_1, \ldots, t_k)$, нужно помнить множество значений S_i, которые могут быть в перестановке на i-ом месте. Так, $S_1 = 1:8, t_1 = 3$ означает, что $r_1 = 4$. Далее $S_2 = 1:3 \cup 5:8, t_2 = 6$ значит, что $r_2 = 8$

Замечание. Если использовать отображение из примера при переборре, то перестановки будут идти в лексикографическом порядке Это значит, что:

$(r_1, \ldots, r_k)$ предшествует $(R_1, \ldots, R_k) \Leftrightarrow$ начала этих перестановок совпадают до i индекса, а далее $r_i < R_i$

Замечание. Очевидно, что если факториальная запись $(t_1, \ldots, t_k)$ лексикографически предшествует другой, то порядок верен и для соответствующих перестановок.

Алгоритм. (Перебор перестановок в лексикографическом порядке)

1. в заданной перестановке $(r_1, \ldots, r_k)$ найдем наибольший суффикс $(r_t, \ldots, r_k)$, в котором элементы расположены по убыванию.

2. выбрать в $(r_t, \ldots, r_k)$ элемент, следующий по велечине после r_{t-1} и поставить его на r_{t-1}. Оставшиеся эелменты, включая r_{t-1} расположить за ним в порядке возрастания.

Пример.

3	4	2	1	7	8	9	5	**6**
3	4	2	1	7	8	**9**	**6**	**5**
3	4	2	1	7	9	5	6	**8**
3	4	2	1	7	9	5	**8**	**6**
3	4	2	1	7	9	6	5	**8**
3	4	2	1	7	9	6	**8**	**5**
3	4	2	1	7	9	8	5	**6**
3	4	2	1	7	**9**	**8**	**6**	**5**
3	4	2	1	8	5	6	7	**9**

Лекция 3: Продолжение

27.09.2023

2.4 Перебор перестановок в лекцикографическом порядке (другой способ)

Алгоритм. Будем брать элементы $(t_1, \ldots, t_k)$ из T_k и сопоставлять им перестановки так, как делали ранее. Переход к следующей перестановке осуществляется путем прибавления единицы к $(t_1, \ldots, t_k)$. (причем последний элемент в $(t_1, \ldots, t_k)$ всегда ноль, т.к. ничего не значит)

Пример. (для P_4)

num	t	p
0	(0, 0, 0, 0)	(1, 2, 3, 4)
1	(0, 0, 1, 0)	(1, 2, 4, 3)
2	(0, 1, 0, 0)	(1, 3, 2, 4)
3	(0, 1, 1, 0)	(1, 3, 4, 2)
4	(0, 2, 0, 0)	(1, 4, 2, 3)
5	(0, 2, 1, 0)	(1, 4, 3, 2)
6	(0, 3, 0, 0)	(2, 1, 3, 4)
$\vdots$	$\vdots$	$\vdots$
23	(3, 2, 1, 0)	(4, 3, 2, 1)

2.5 Перебор с минимальным изменением

На каждой итерации будем менять только два соседних элемента. Для этого необходимо:

- берем последний элемент в перестановке и меняем его с соседом до тех пор, пока элемент не дойдет до начала.
- когда этот элемент оказался в начале, мы меняем у него направление: теперь он будет менятся с соседом справ, а элемент, который оказался на последней позиции, делает 1 шаг (и так каждый раз, когда наш первый элемент меняет направление). Такие действия применяются ко всем элементам в перестановке.

Алгоритм. Кроме самой перестановки p и ее номера t (на этот раз младший разряд в номере — последний), будем хранить массив d, в котором будем хранить направление движения элементов. Если элемент движется вправо, то d[i] = +, если влево, то d[i] = -. Начальное значение d[i] = - для всех i.

Также храним j где будем записывать индекс элемента в t, в котором значение увеличилось.

1. Прибавляем 1 к t
2. Определяем номер разряда в котором значение увеличивается на 1, записываем в j
3. $\forall i \in [1, n] : i > j$, меняем $d_i = -d_i$.
4. j (не номер, именно такой элемент) меняем с соседом слева если $d_j = -$, и с соседом справа, если $d_j = +$.

Пример.

num	t	d	p	j
0	0000	$- - --$	1234	-
1	0001	$- - --$	1243	4
2	0002	$- - --$	1423	4
3	0003	$- - --$	4123	4
4	0010	$- - -+$	4132	3
5	0011	$- - -+$	1432	4
6	0012	$- - -+$	1342	4
⋮	⋮	⋮	⋮	⋮
19	0113	$- - ++$	4231	4
20	0120	$- - ++$	4213	3
21	0121	$- - ++$	2413	4
22	0122	$- - ++$	2143	4
23	0123	$- - ++$	2134	4
24	1000	$- + --$	—	1

2.6 Сочетания и бином Ньютона

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

Свойства. (Свойства сочетаний)

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

Доказательство. Доказывается путем подстановки непосредственно в формулу, или можно рассматривать пути на целочисленной решетке. □

Теорема 3. (Бином Ньютона)

$$\forall a, b \in \mathbb{R}, n \in \mathbb{N}_0 : (a+b)^n = \sum_{k=0}^{n} C_n^k a^k b^{n-k}$$

Доказательство. (По индкуции)

1. База $n = 1$ очевидна.
2. индкуционый переход $n - 1 \to n$:

$$(a+b)^n = (a+b)(a+b)^{n-1} = a(a+b)^{n-1} + b(a+b)^{n-1} =$$

$$= a \cdot \sum_{k=0}^{n-1} C_{n-1}^k a^k b^{n-1-k} + b \cdot \sum_{k=0}^{n-1} C_{n-1}^k a^k b^{n-1-k} =$$

$$= \sum_{k=0}^{n-1} C_{n-1}^k a^{k+1} b^{n-1-k} + \sum_{k=0}^{n-1} C_{n-1}^k a^k b^{n-k} =$$

$$= \sum_{k=1}^{n} C_{n-1}^{k-1} a^k b^{n-k} + \sum_{k=0}^{n-1} C_{n-1}^k a^k b^{n-k} =$$

$$= a^n + \sum_{k=1}^{n-1} C_{n-1}^{k-1} a^k b^{n-k} + b^n + \sum_{k=1}^{n-1} C_{n-1}^k a^k b^{n-k} =$$

$$= a^n + b^n + \sum_{k=1}^{n-1} (C_{n-1}^k + C_{n-1}^{k-1}) a^k b^{n-k} = (a+b)^n$$

□

2.7 Перебор сочетаний с хорошей нумерацией

Для того чтобы присваивать номер сочетанию, будем рассматривать сочетание как вектор из нулей и единиц: если элемент взяли — единица, иначе — ноль.

Пример. Вектору $b = (1, 0, 1, 1, 1, 0, 0, 0)$ соответствует сочетание 1345.

Алгоритм. определяем номер рекурсивно:

$$num(b[1:n-1], m) = \begin{cases} num(b[1:n-1], m), & \text{если} b[n] = 0, \\ num(b[1:n-1], m-1), & \text{если} b[n] = 1, \end{cases}$$

Где m — кол-во единиц.

Пример. Рассмотрим $b = (1, 0, 1, 1, 1, 0, 0, 0), m = 4$:

$$\begin{aligned} num(b, m) =& C_6^4 + num(b[1:n-1], 3) = \\ =& C_6^4 + C_4^3 + num(b[1:n-2], 3) = \\ =& C_6^4 + C_4^3 + num(b[1:n-3], 2) = \\ =& C_6^4 + C_4^3 + num(b[1:n-4], 2) = \\ =& C_6^4 + C_4^3 + C_2^2 + num(b[1:n-5], 1) = \\ =& C_6^4 + C_4^3 + C_2^2 + 0 = 15 + 4 + 1 = 20 \end{aligned}$$