Tuples and Permutations

Vladimir Podolskii

Computer Science Department, Higher School of Economics

Outline

Number of Tuples

Set Language for Tuples

Licence Plates

Tuples with Restrictions

Permutations

Number of Passwords

Problem

How many different 5-symbol passwords can we create using lower case Latin letters only? (the size of the alphabet is 26)

Number of Passwords

Problem

How many different 5-symbol passwords can we create using lower case Latin letters only? (the size of the alphabet is 26)

- It turns out that the rule of product is all we need to solve this problem

Number of Passwords

Problem

How many different 5-symbol passwords can we create using lower case Latin letters only? (the size of the alphabet is 26)

- It turns out that the rule of product is all we need to solve this problem
- But we need to do it step by step

Number of Passwords

- Let's start with 1 letter password

*

Number of Passwords

- Let's start with 1 letter password
- Clearly, then the answer is 26

*

Number of Passwords

- Let's start with 1 letter password
- Clearly, then the answer is 26
- What about two letters?

$*$ $*$

Number of Passwords

- Let's start with 1 letter password
- Clearly, then the answer is 26
- What about two letters?
- Then we can choose both letters in 26 ways

26	26
*	*

Number of Passwords

- Let's start with 1 letter password
- Clearly, then the answer is 26
- What about two letters?
- Then we can choose both letters in 26 ways
- Use the rule of product: the answer is 676

$$26 \times 26 = 676$$

$$* \quad *$$

Number of Passwords

- Let's move on to the case of 3 letters

$*\quad *\quad *$

Number of Passwords

- Let's move on to the case of 3 letters
- We already know that we can choose the first two letters in 676 ways

676

$\boxed{*\quad *}\quad *$

Number of Passwords

- Let's move on to the case of 3 letters
- We already know that we can choose the first two letters in 676 ways
- We apply the rule of product again!

$$\begin{array}{ccc} \multicolumn{2}{c}{676} & \times\ 26 \\ \boxed{* \quad *} & & * \end{array}$$

Number of Passwords

- Let's move on to the case of 3 letters
- We already know that we can choose the first two letters in 676 ways
- We apply the rule of product again!
- And the answer is 17 576

$$\underbrace{676}_{\boxed{*\quad *}} \times \underbrace{26}_{*} = 17\,576$$

Number of Passwords

- Now we can proceed in the same way

$$26 \times 26 \times 26$$

$$* \quad * \quad * \quad * \quad *$$

Number of Passwords

- Now we can proceed in the same way

$$26 \times 26 \times 26 \times 26$$

$$* \quad * \quad * \quad * \quad *$$

Number of Passwords

- Now we can proceed in the same way

$$26 \times 26 \times 26 \times 26 \times 26$$

$$* \quad * \quad * \quad * \quad *$$

Number of Passwords

- Now we can proceed in the same way
- And the answer is 11 881 376

$$26 \times 26 \times 26 \times 26 \times 26 = 11\,881\,376$$

$$* \quad * \quad * \quad * \quad *$$

Number of Tuples

Problem

Suppose we have a set of n symbols. How many different sequences of length k we can form out of these symbols?

These sequences are usually called tuples

Number of Tuples

Problem

Suppose we have a set of n symbols. How many different sequences of length k we can form out of these symbols?

These sequences are usually called tuples

- Can apply the same argument

Number of Tuples

Problem

Suppose we have a set of n symbols. How many different sequences of length k we can form out of these symbols?

These sequences are usually called tuples

- Can apply the same argument
- There are n possibilities to pick the first letter

Number of Tuples

Problem

Suppose we have a set of n symbols. How many different sequences of length k we can form out of these symbols?

These sequences are usually called tuples

- Can apply the same argument
- There are n possibilities to pick the first letter
- Each next letter multiplies the number of sequences by n

Number of Tuples

Problem

Suppose we have a set of n symbols. How many different sequences of length k we can form out of these symbols?

These sequences are usually called tuples

- Can apply the same argument
- There are n possibilities to pick the first letter
- Each next letter multiplies the number of sequences by n
- Thus the answer is a product of n by itself k times, that is n^k

Outline

Number of Tuples

Set Language for Tuples

Licence Plates

Tuples with Restrictions

Permutations

Cartesian Product

- There is a set notation for tuples

Cartesian Product

- There is a set notation for tuples
- Suppose we are given two sets A and B

Cartesian Product

- There is a set notation for tuples
- Suppose we are given two sets A and B
- By $A \times B$ we denote the set of all pairs (a, b), where $a \in A$ and $b \in B$

Cartesian Product

- There is a set notation for tuples
- Suppose we are given two sets A and B
- By $A \times B$ we denote the set of all pairs (a, b), where $a \in A$ and $b \in B$
- $A \times B$ is called Cartesian product of sets A and B

Cartesian Product

- There is a set notation for tuples
- Suppose we are given two sets A and B
- By $A \times B$ we denote the set of all pairs (a, b), where $a \in A$ and $b \in B$
- $A \times B$ is called Cartesian product of sets A and B
- If A and B are finite, the number of elements in $A \times B$ is equal to $|A| \cdot |B|$

Cartesian Product

- There is a set notation for tuples
- Suppose we are given two sets A and B
- By $A \times B$ we denote the set of all pairs (a, b), where $a \in A$ and $b \in B$
- $A \times B$ is called Cartesian product of sets A and B
- If A and B are finite, the number of elements in $A \times B$ is equal to $|A| \cdot |B|$
- That is, $|A \times B| = |A| \cdot |B|$

Cartesian Product

- There is a set notation for tuples
- Suppose we are given two sets A and B
- By $A \times B$ we denote the set of all pairs (a, b), where $a \in A$ and $b \in B$
- $A \times B$ is called Cartesian product of sets A and B
- If A and B are finite, the number of elements in $A \times B$ is equal to $|A| \cdot |B|$
- That is, $|A \times B| = |A| \cdot |B|$
- This is just the rule of product

Cartesian Product

- More generally, suppose we are given sets $A_1, A_2, \ldots, A_k$

Cartesian Product

- More generally, suppose we are given sets $A_1, A_2, \ldots, A_k$
- By $A_1 \times A_2 \times \ldots \times A_k$ we denote the set of all tuples $(a_1, a_2, \ldots, a_k)$, where $a_1 \in A_1$, $a_2 \in A_2$ and so on

Cartesian Product

- More generally, suppose we are given sets $A_1, A_2, \ldots, A_k$
- By $A_1 \times A_2 \times \ldots \times A_k$ we denote the set of all tuples $(a_1, a_2, \ldots, a_k)$, where $a_1 \in A_1$, $a_2 \in A_2$ and so on
- $A_1 \times A_2 \times \ldots \times A_k$ is called Cartesian product of sets $A_1, A_2, \ldots, A_k$

Cartesian Product

- More generally, suppose we are given sets $A_1, A_2, \ldots, A_k$
- By $A_1 \times A_2 \times \ldots \times A_k$ we denote the set of all tuples $(a_1, a_2, \ldots, a_k)$, where $a_1 \in A_1$, $a_2 \in A_2$ and so on
- $A_1 \times A_2 \times \ldots \times A_k$ is called Cartesian product of sets $A_1, A_2, \ldots, A_k$
- When $A_1 = A_2 = \ldots = A_k$ it is convenient to shorten the notation to A^k

Cartesian Product

- More generally, suppose we are given sets $A_1, A_2, \dots, A_k$
- By $A_1 \times A_2 \times \dots \times A_k$ we denote the set of all tuples $(a_1, a_2, \dots, a_k)$, where $a_1 \in A_1$, $a_2 \in A_2$ and so on
- $A_1 \times A_2 \times \dots \times A_k$ is called Cartesian product of sets $A_1, A_2, \dots, A_k$
- When $A_1 = A_2 = \dots = A_k$ it is convenient to shorten the notation to A^k
- That is, the set of tuples in which each coordinate is taken from the same set is denoted by A^k

Cartesian Product

- We have seen in the previous video that for finite A

$$|A^k| = |A|^k$$

Cartesian Product

- We have seen in the previous video that for finite A

$$|A^k| = |A|^k$$

- In fact, completely analogously we can show that for finite $A_1, A_2, \ldots, A_k$

$$|A_1 \times A_2 \times \ldots \times A_k| = |A_1| \cdot |A_2| \cdot \ldots \cdot |A_k|$$

Cartesian Product

- We have seen in the previous video that for finite A

$$|A^k| = |A|^k$$

- In fact, completely analogously we can show that for finite $A_1, A_2, \ldots, A_k$

$$|A_1 \times A_2 \times \ldots \times A_k| = |A_1| \cdot |A_2| \cdot \ldots \cdot |A_k|$$

- We will see this in the next video

Tuples in Data Science

- We will see tuples a lot in Data Science

Tuples in Data Science

- We will see tuples a lot in Data Science
- Suppose your data have features ‘Car model’, ‘Phone model’, Occupation’

Tuples in Data Science

- We will see tuples a lot in Data Science
- Suppose your data have features ‘Car model’, ‘Phone model’, Occupation’
- Then each person is encoded in your data as a tuple

Tuples in Data Science

- We will see tuples a lot in Data Science
- Suppose your data have features ‘Car model’, ‘Phone model’, Occupation’
- Then each person is encoded in your data as a tuple
- for example
  (Chevrolet Suburban, BlackBerry Torch 9810, Manager)
  might be your data entry

Outline

Number of Tuples

Set Language for Tuples

Licence Plates

Tuples with Restrictions

Permutations

Licence Plates

wikimedia.org

- Now we are ready to get back to our motivating example

Licence Plates

wikimedia.org

- Now we are ready to get back to our motivating example
- Russian license plate: 3 digits, 3 letters; 78 is a regional code

Licence Plates

wikimedia.org

- Now we are ready to get back to our motivating example
- Russian license plate: 3 digits, 3 letters; 78 is a regional code
- We have: 10 options for digits, 12 options for letters (only those Cyrillic letters that are similar to Latin ones are used)

Licence Plates

wikimedia.org

- Now we are ready to get back to our motivating example
- Russian license plate: 3 digits, 3 letters; 78 is a regional code
- We have: 10 options for digits, 12 options for letters (only those Cyrillic letters that are similar to Latin ones are used)
- How many plates are there for one region?

Licence Plates

wikimedia.org

- Each digit can be chosen in 10 ways

Licence Plates

wikimedia.org

- Each digit can be chosen in 10 ways
- Thus a sequence of digits can be chosen in $10 \times 10 \times 10 = 1\,000$ ways

Licence Plates

wikimedia.org

- Each digit can be chosen in 10 ways
- Thus a sequence of digits can be chosen in $10 \times 10 \times 10 = 1\,000$ ways
- Each letter can be chosen in 12 ways

Licence Plates

wikimedia.org

- Each digit can be chosen in 10 ways
- Thus a sequence of digits can be chosen in $10 \times 10 \times 10 = 1\,000$ ways
- Each letter can be chosen in 12 ways
- Thus a sequence of three letters can be chosen in $12 \times 12 \times 12 = 1\,728$ ways

Licence Plates

wikimedia.org

- Each digit can be chosen in 10 ways
- Thus a sequence of digits can be chosen in $10 \times 10 \times 10 = 1\,000$ ways
- Each letter can be chosen in 12 ways
- Thus a sequence of three letters can be chosen in $12 \times 12 \times 12 = 1\,728$ ways
- Overall, there are 1 728 000 license plates for a region

Licence Plates

wikimedia.org

- There are 1 728 000 license plates for a region
- Is it enough?

Licence Plates

wikimedia.org

- There are 1 728 000 license plates for a region
- Is it enough?
- No, it's not: for example, there are about 5 600 000 vehicles in Moscow (as of 2016)

Licence Plates

wikimedia.org

- There are 1 728 000 license plates for a region
- Is it enough?
- No, it's not: for example, there are about 5 600 000 vehicles in Moscow (as of 2016)
- Does it mean that there are vehicles with identical license plates?

Licence Plates

wikimedia.org

- No, several regional codes were introduced for the same region

Licence Plates

wikimedia.org

- No, several regional codes were introduced for the same region
- But this required introduction of three-digit regional codes

Outline

Number of Tuples

Set Language for Tuples

Licence Plates

Tuples with Restrictions

Permutations

Tuples with Restrictions

- We have shown how using the rule of product we can compute the number of tuples or a certain length of a fixed set of symbols

Tuples with Restrictions

- We have shown how using the rule of product we can compute the number of tuples or a certain length of a fixed set of symbols
- But the rule of product can also give us other things too

Numbers with exactly one digit 7

Problem

How many integer numbers between 0 and 9999 are there that have exactly one 7 digit?

Numbers with exactly one digit 7

Problem

How many integer numbers between 0 and 9999 are there that have exactly one 7 digit?

- Numbers between 0 and 9999 are sequences of digits of length 4

Numbers with exactly one digit 7

Problem

How many integer numbers between 0 and 9999 are there that have exactly one 7 digit?

- Numbers between 0 and 9999 are sequences of digits of length 4
- Three digital numbers correspond to sequences starting with 0

Numbers with exactly one 7 digit

$$* \quad * \quad * \quad *$$

- We can place the unique 7 at any of four positions

Numbers with exactly one 7 digit

$$* \quad * \quad * \quad *$$

- We can place the unique 7 at any of four positions
- This gives us 4 cases; if we compute the number of sequences in all four cases, we can get the answer by the rule of sum

Numbers with exactly one 7 digit

$$* \quad 7 \quad * \quad *$$

- We can place the unique 7 at any of four positions
- This gives us 4 cases; if we compute the number of sequences in all four cases, we can get the answer by the rule of sum
- Consider one of the cases

Numbers with exactly one 7 digit

$$* \quad 7 \quad * \quad *$$

- We can place the unique 7 at any of four positions
- This gives us 4 cases; if we compute the number of sequences in all four cases, we can get the answer by the rule of sum
- Consider one of the cases
- Each of other three digits can be picked out of 9 options! (digit 7 is forbidden)

Numbers with exactly one 7 digit

$$* \quad 7 \quad * \quad *$$

- Thus there are $9 \times 9 \times 9 = 729$ sequences in this case

Numbers with exactly one 7 digit

$$* \quad * \quad * \quad *$$

- Thus there are $9 \times 9 \times 9 = 729$ sequences in this case
- And in all other cases as well!

Numbers with exactly one 7 digit

$*\quad *\quad *\quad *$

- Thus there are $9 \times 9 \times 9 = 729$ sequences in this case
- And in all other cases as well!
- There are 4 cases, so there are $4 \times 729 = 2\,916$ four numbers below 10 000 with exactly one digit 7

Numbers with exactly one 7 digit

$$* \quad * \quad * \quad *$$

- Thus there are $9 \times 9 \times 9 = 729$ sequences in this case
- And in all other cases as well!
- There are 4 cases, so there are $4 \times 729 = 2\,916$ four numbers below 10 000 with exactly one digit 7
- This is below 1/3, but above 1/4 of all four digit numbers

Numbers with exactly one 7 digit

$$* \quad * \quad * \quad *$$

- Thus there are $9 \times 9 \times 9 = 729$ sequences in this case
- And in all other cases as well!
- There are 4 cases, so there are $4 \times 729 = 2\,916$ four numbers below 10 000 with exactly one digit 7
- This is below 1/3, but above 1/4 of all four digit numbers
- This is an estimation of the probability to get exactly one digit 7 if we pick a number below 10 000 “randomly”

Outline

Number of Tuples

Set Language for Tuples

Licence Plates

Tuples with Restrictions

Permutations

Permutations

- We have discussed how to count the number of tuples

Permutations

- We have discussed how to count the number of tuples
- Now we are ready to proceed to the second standard combinatorial setting: permutations

Permutations

Problem

Suppose we have a set of n symbols. How many different sequences of length k we can form out of these symbols if we are not allowed to use the same symbol twice?

- Tuples of length k without repetitions are called k-permutations

Permutations

Problem

Suppose we have a set of n symbols. How many different sequences of length k we can form out of these symbols if we are not allowed to use the same symbol twice?

- Tuples of length k without repetitions are called k-permutations
- Observe that if $n < k$, then there are no k-permutations: there are simply not enough different letters

Permutations

Problem

Suppose we have a set of n symbols. How many different sequences of length k we can form out of these symbols if we are not allowed to use the same symbol twice?

- Tuples of length k without repetitions are called k-permutations
- Observe that if $n < k$, then there are no k-permutations: there are simply not enough different letters
- So it is enough to solve the problem for the case $k \leq n$

Permutations

$$\begin{array}{ccccc} 1 & 2 & 3 & \ldots & k \\ * & * & * & \cdots & * \end{array}$$

- Let us apply rule of product

Permutations

$$\begin{array}{ccccc} 1 & 2 & 3 & \ldots & k \\ * & * & * & \cdots & * \\ n & & & & \end{array}$$

- Let us apply rule of product
- The first symbol can be picked in n ways

Permutations

$$\begin{array}{ccccc} 1 & 2 & 3 & \ldots & k \\ * & * & * & \cdots & * \\ n & & & & \end{array}$$

- Let us apply rule of product
- The first symbol can be picked in n ways
- How many choices there are for the second symbol?

Permutations

$$\begin{array}{ccccc} 1 & 2 & 3 & \ldots & k \\ * & * & * & \cdots & * \\ n & & & & \end{array}$$

- Let us apply rule of product
- The first symbol can be picked in n ways
- How many choices there are for the second symbol?
- We can place anything there, except for the symbol on the first place

Permutations

$$\begin{array}{ccccc} 1 & 2 & 3 & \ldots & k \\ * & * & * & \cdots & * \\ n & n-1 & & & \end{array}$$

- Let us apply rule of product
- The first symbol can be picked in n ways
- How many choices there are for the second symbol?
- We can place anything there, except for the symbol on the first place
- Symbol on the first place might be arbitrary, but whatever it is there are $n-1$ choices for the second symbol!

Permutations

$$\begin{array}{ccccc} 1 & 2 & 3 & \ldots & k \\ * & * & * & \cdots & * \\ n & n-1 & & & \end{array}$$

- So we can pick the first and the second symbol in $n \times (n-1)$ ways

Permutations

$$\begin{array}{ccccc} 1 & 2 & 3 & \ldots & k \\ * & * & * & \cdots & * \\ n & n{-}1 & n{-}2 & & \end{array}$$

- So we can pick the first and the second symbol in $n{\times}(n{-}1)$ ways
- Now, the third symbol can be picked among $n{-}2$ options: all except the symbols in the first and the second position

Permutations

$$\begin{array}{ccccc} 1 & 2 & 3 & \dots & k \\ * & * & * & \cdots & * \\ n & n-1 & n-2 & \dots & \end{array}$$

- So we can pick the first and the second symbol in $n \times (n-1)$ ways
- Now, the third symbol can be picked among $n-2$ options: all except the symbols in the first and the second position
- And so on; for each next symbol we have one less option

Permutations

$$\begin{array}{ccccc} 1 & 2 & 3 & \ldots & k \\ * & * & * & \cdots & * \\ n & n-1 & n-2 & \ldots & n-k+1 \end{array}$$

- So we can pick the first and the second symbol in $n \times (n-1)$ ways
- Now, the third symbol can be picked among $n-2$ options: all except the symbols in the first and the second position
- And so on; for each next symbol we have one less option
- In the end for the last object we have $n-k+1$ options

Permutations

$$\begin{array}{ccccc} 1 & 2 & 3 & \ldots & k \\ * & * & * & \cdots & * \\ n & n-1 & n-2 & \ldots & n-k+1 \end{array}$$

- Overall we have $n \times (n-1) \times \ldots \times (n-k+1)$ k-permutations

Permutations

$$\begin{array}{ccccc} 1 & 2 & 3 & \ldots & k \\ * & * & * & \cdots & * \\ n & n-1 & n-2 & \ldots & n-k+1 \end{array}$$

- Overall we have $n \times (n-1) \times \ldots \times (n-k+1)$ k-permutations
- Convenient notation: $n! = 1 \times 2 \times \ldots \times n$; this number is called factorial of n

Permutations

$$\begin{array}{ccccc} 1 & 2 & 3 & \dots & k \\ * & * & * & \cdots & * \\ n & n-1 & n-2 & \dots & n-k+1 \end{array}$$

- Overall we have $n \times (n-1) \times \ldots \times (n-k+1)$ k-permutations
- Convenient notation: $n! = 1 \times 2 \times \ldots \times n$; this number is called factorial of n
- In this notation the number of k-permutations of n symbols of length k looks nicer: it is $n!/(n-k)!$

Permutations

$$\begin{array}{ccccc} 1 & 2 & 3 & \ldots & k \\ * & * & * & \cdots & * \\ n & n-1 & n-2 & \ldots & n-k+1 \end{array}$$

- Overall we have $n \times (n-1) \times \ldots \times (n-k+1)$ k-permutations
- Convenient notation: $n! = 1 \times 2 \times \ldots \times n$; this number is called factorial of n
- In this notation the number of k-permutations of n symbols of length k looks nicer: it is $n!/(n-k)!$
- What if $n-k=0$?

Permutations

$$\begin{array}{ccccc} 1 & 2 & 3 & \dots & k \\ * & * & * & \cdots & * \\ n & n-1 & n-2 & \dots & n-k+1 \end{array}$$

- Overall we have $n \times (n-1) \times \dots \times (n-k+1)$ k-permutations
- Convenient notation: $n! = 1 \times 2 \times \dots \times n$; this number is called factorial of n
- In this notation the number of k-permutations of n symbols of length k looks nicer: it is $n!/(n-k)!$
- What if $n-k=0$? Convention: $0!=1$

Permutations

Problem

In how many orders can we place n books on the shelf?

Permutations

Problem

In how many orders can we place n books on the shelf?

- Each book is a symbol

Permutations

Problem

In how many orders can we place n books on the shelf?

- Each book is a symbol
- We need to count n-permutations of n symbols; these are called permutations

Permutations

Problem

In how many orders can we place n books on the shelf?

- Each book is a symbol
- We need to count n-permutations of n symbols; these are called permutations
- By the previous result there are $n!$ of them

Conclusion

- We have discussed two standard settings: tuples and permutations

Conclusion

- We have discussed two standard settings: tuples and permutations
- They help in many cases

Conclusion

- We have discussed two standard settings: tuples and permutations
- They help in many cases
- But they do not cover all that is needed

Conclusion

- We have discussed two standard settings: tuples and permutations
- They help in many cases
- But they do not cover all that is needed
- One more standard setting in the next lesson