Recursive Counting

Vladimir Podolskii

Computer Science Department, Higher School of Economics

Outline

Number of Paths

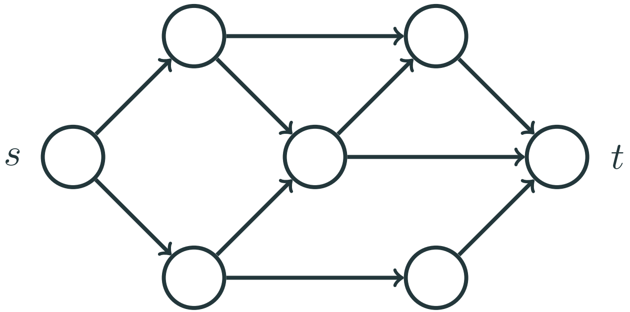
Rule of Product

Back to Recursive Counting

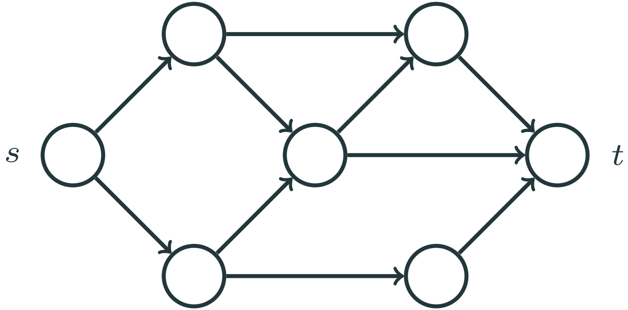
Number of Paths

Problem

Suppose there are several points connected by arrows. There is a starting point s (called source) and a final point t (called sink). How many different ways are there to get from s to t?

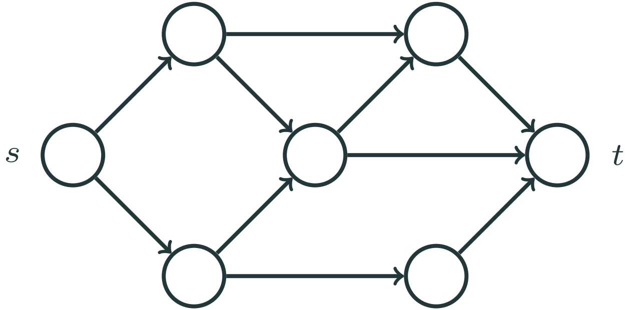


Number of Paths



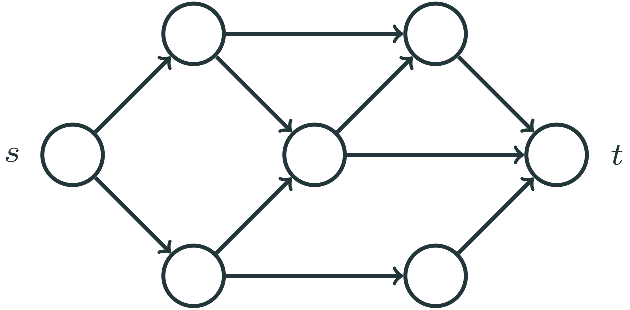
- There are several various paths; how not to miss anything when counting?

Number of Paths



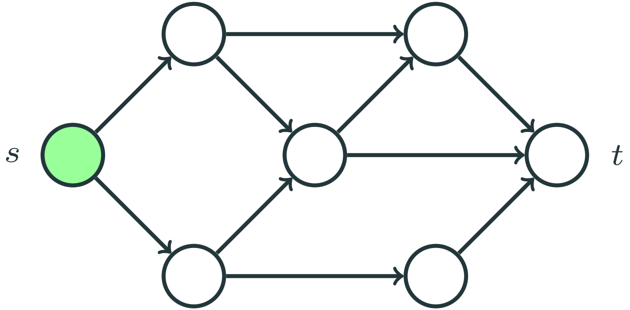
- There are several various paths; how not to miss anything when counting?
- We can count them recursively: for each node count the number of paths from s to this node

Number of Paths



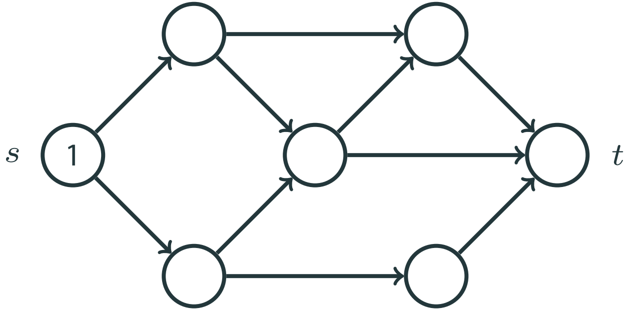
- There are several various paths; how not to miss anything when counting?
- We can count them recursively: for each node count the number of paths from s to this node
- We use the rule of sum!

Number of Paths



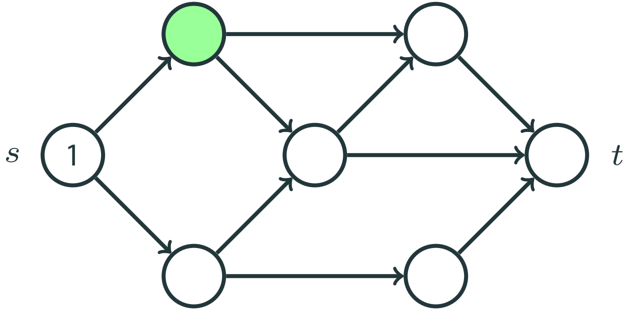
- There are several various paths; how not to miss anything when counting?
- We can count them recursively: for each node count the number of paths from s to this node
- We use the rule of sum!

Number of Paths



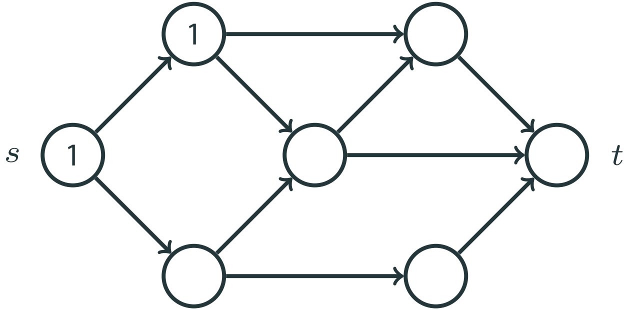
- There are several various paths; how not to miss anything when counting?
- We can count them recursively: for each node count the number of paths from s to this node
- We use the rule of sum!

Number of Paths



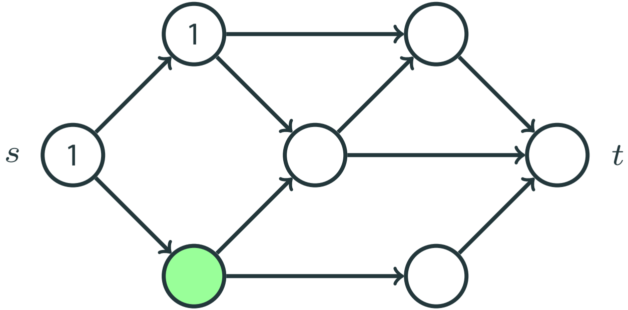
- There are several various paths; how not to miss anything when counting?
- We can count them recursively: for each node count the number of paths from s to this node
- We use the rule of sum!

Number of Paths



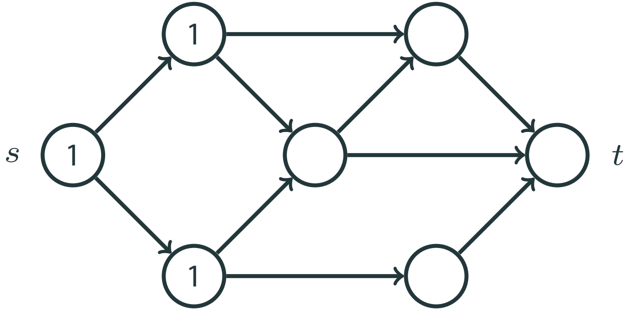
- There are several various paths; how not to miss anything when counting?
- We can count them recursively: for each node count the number of paths from s to this node
- We use the rule of sum!

Number of Paths



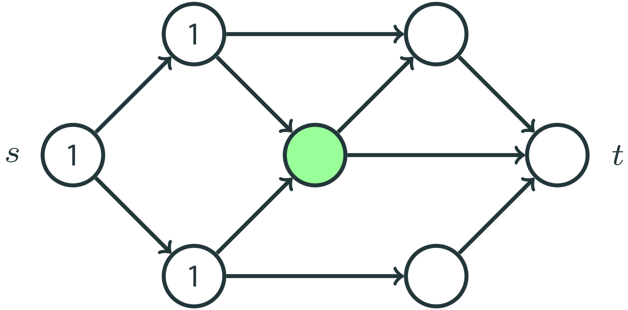
- There are several various paths; how not to miss anything when counting?
- We can count them recursively: for each node count the number of paths from s to this node
- We use the rule of sum!

Number of Paths



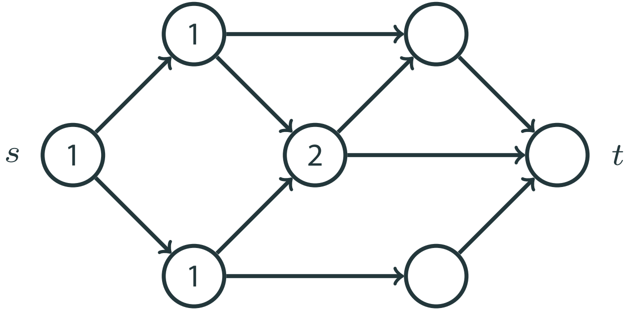
- There are several various paths; how not to miss anything when counting?
- We can count them recursively: for each node count the number of paths from s to this node
- We use the rule of sum!

Number of Paths



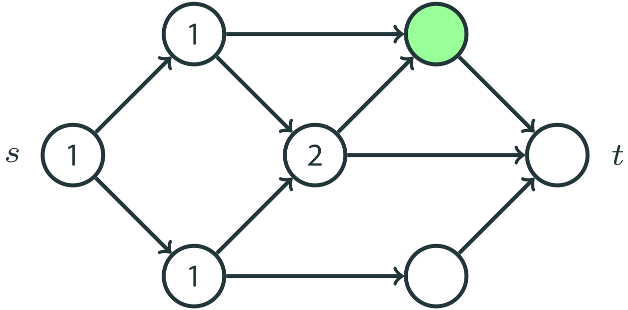
- There are several various paths; how not to miss anything when counting?
- We can count them recursively: for each node count the number of paths from s to this node
- We use the rule of sum!

Number of Paths



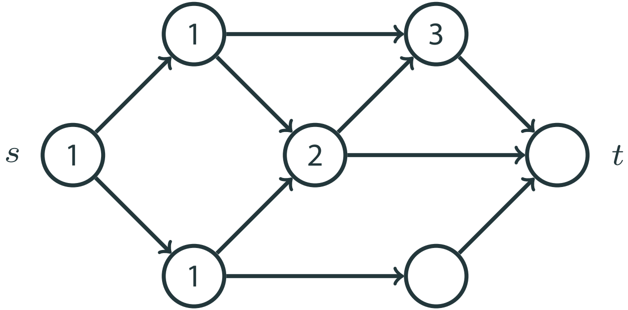
- There are several various paths; how not to miss anything when counting?
- We can count them recursively: for each node count the number of paths from s to this node
- We use the rule of sum!

Number of Paths



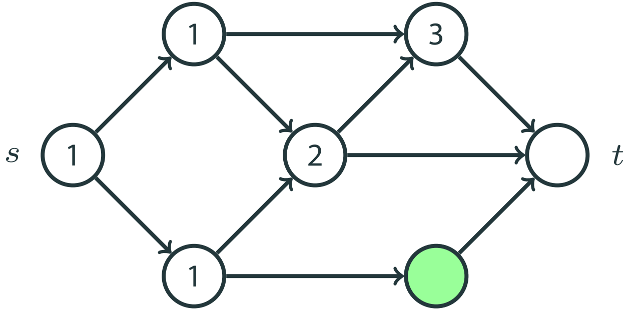
- There are several various paths; how not to miss anything when counting?
- We can count them recursively: for each node count the number of paths from s to this node
- We use the rule of sum!

Number of Paths



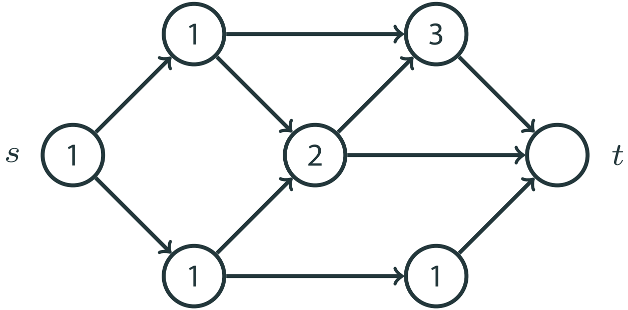
- There are several various paths; how not to miss anything when counting?
- We can count them recursively: for each node count the number of paths from s to this node
- We use the rule of sum!

Number of Paths



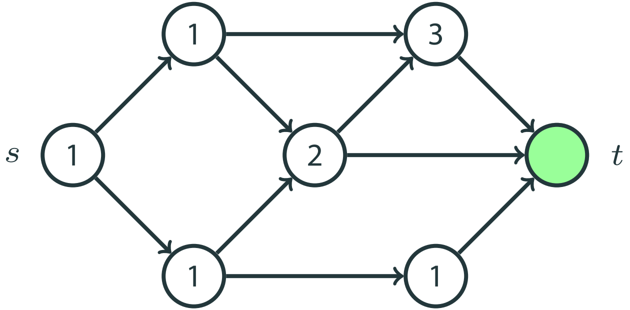
- There are several various paths; how not to miss anything when counting?
- We can count them recursively: for each node count the number of paths from s to this node
- We use the rule of sum!

Number of Paths



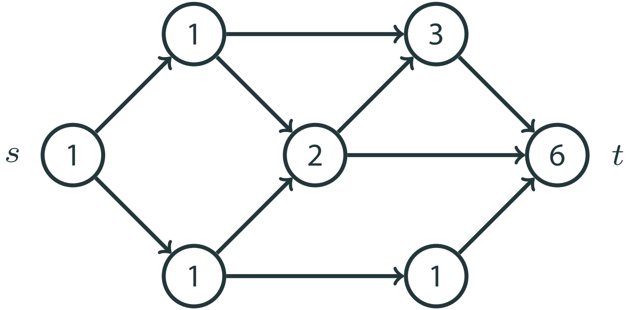
- There are several various paths; how not to miss anything when counting?
- We can count them recursively: for each node count the number of paths from s to this node
- We use the rule of sum!

Number of Paths



- There are several various paths; how not to miss anything when counting?
- We can count them recursively: for each node count the number of paths from s to this node
- We use the rule of sum!

Number of Paths



- There are several various paths; how not to miss anything when counting?
- We can count them recursively: for each node count the number of paths from s to this node
- We use the rule of sum!

Outline

Number of Paths

Rule of Product

Back to Recursive Counting

Rule of Product

Rule of Product

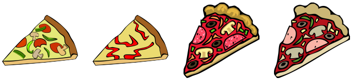
If there are k object of the first type and there are n object of the second type, then there are $k \times n$ pairs of objects, the first of the first type and the second of the second type

Rule of Product

Rule of Product

If there are k object of the first type and there are n object of the second type, then there are $k \times n$ pairs of objects, the first of the first type and the second of the second type

Pizza options



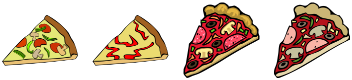
Soda options

Rule of Product

Rule of Product

If there are k object of the first type and there are n object of the second type, then there are $k \times n$ pairs of objects, the first of the first type and the second of the second type

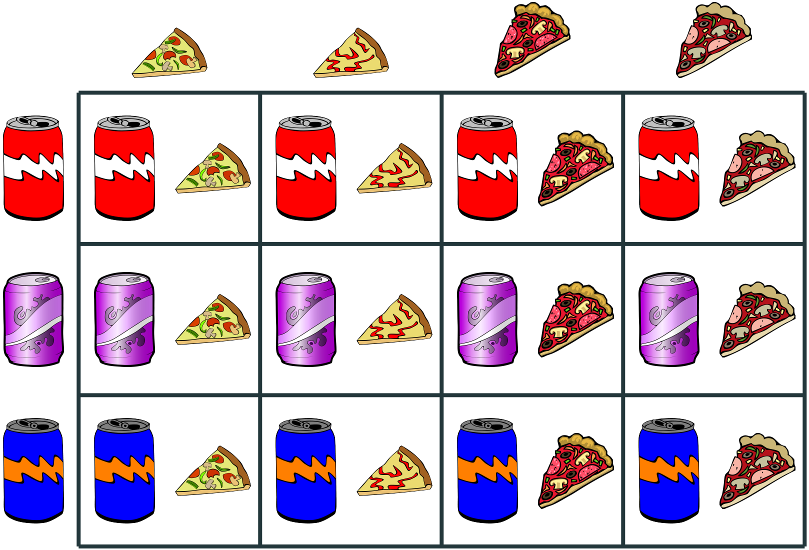
Pizza options



Soda options

$4 \times 3 = 12$ combo options

List of All Combo Options



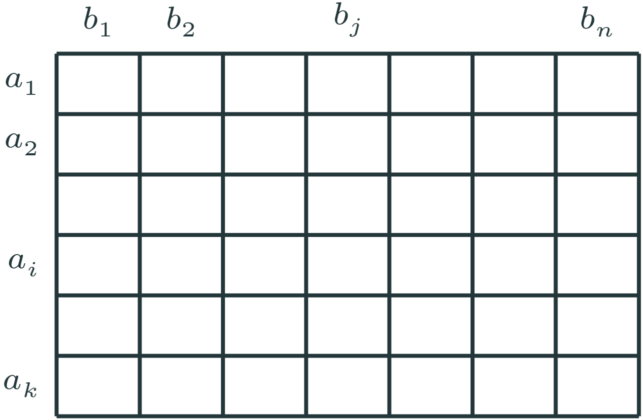
Rule of Product in the Set Language

Rule of Product

If there is a finite set A and a finite set B, then there are $|A| \times |B|$ pairs of objects, the first from A and the second from B

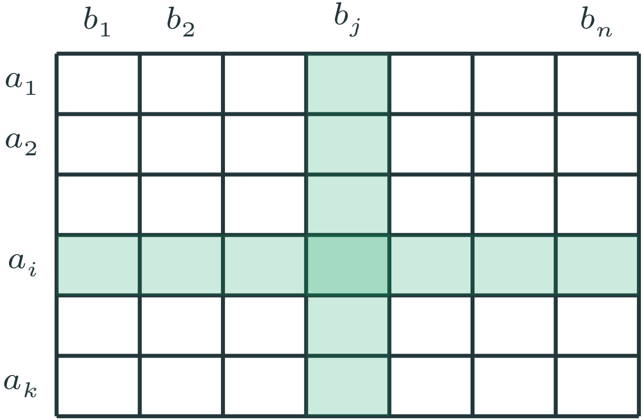
Why the Rule of Product is True?

$A = \{a_1, \dots, a_k\}$
$B = \{b_1, \dots, b_n\}$



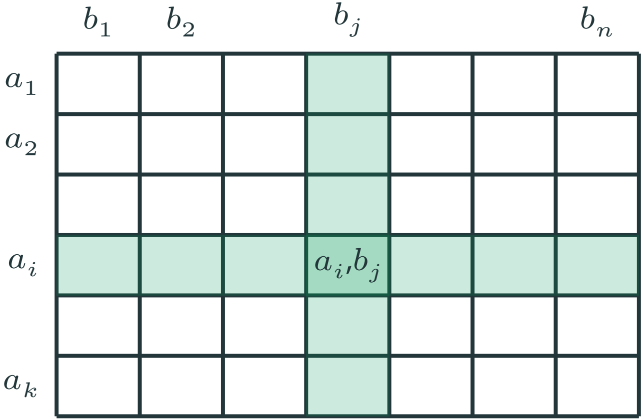
Why the Rule of Product is True?

$$A = \{a_1, \dots, a_k\}$$
$$B = \{b_1, \dots, b_n\}$$



Why the Rule of Product is True?

$$A = \{a_1, \dots, a_k\}$$
$$B = \{b_1, \dots, b_n\}$$



Why the Rule of Product is True?

$A = \{a_1, \dots, a_k\}$
$B = \{b_1, \dots, b_n\}$

	b_1	b_2		b_j			b_n
a_1							
a_2							
a_i				a_i, b_j			
a_k							

There are as many pairs as cells in this table

Outline

Number of Paths

Rule of Product

Back to Recursive Counting

Rule of Product as the Number of Paths

Rule of Product

If there is a finite set A and a finite set B, then there are $|A| \times |B|$ pairs of objects, the first from A and the second from B

Rule of Product as the Number of Paths

Rule of Product

If there is a finite set A and a finite set B, then there are $|A| \times |B|$ pairs of objects, the first from A and the second from B

Can we express this counting rule in terms of counting the number of paths?

Rule of Product as the Number of Paths

Rule of Product

If there is a finite set A and a finite set B, then there are $|A| \times |B|$ pairs of objects, the first from A and the second from B

Rule of Product as the Number of Paths

Rule of Product

If there is a finite set A and a finite set B, then there are $|A| \times |B|$ pairs of objects, the first from A and the second from B

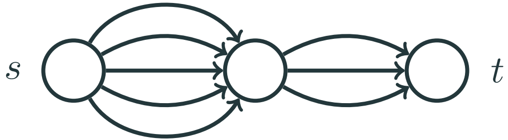
Consider for simplicity $|A| = 5$ and $|B| = 3$

Rule of Product as the Number of Paths

Rule of Product

If there is a finite set A and a finite set B, then there are $|A| \times |B|$ pairs of objects, the first from A and the second from B

Consider for simplicity $|A| = 5$ and $|B| = 3$

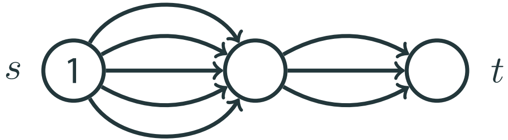


Rule of Product as the Number of Paths

Rule of Product

If there is a finite set A and a finite set B, then there are $|A| \times |B|$ pairs of objects, the first from A and the second from B

Consider for simplicity $|A| = 5$ and $|B| = 3$

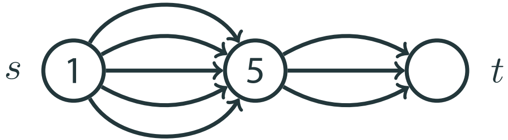


Rule of Product as the Number of Paths

Rule of Product

If there is a finite set A and a finite set B, then there are $|A| \times |B|$ pairs of objects, the first from A and the second from B

Consider for simplicity $|A| = 5$ and $|B| = 3$



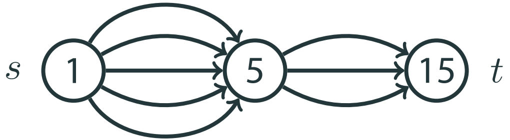
$$1 + 1 + 1 + 1 + 1 = 5$$

Rule of Product as the Number of Paths

Rule of Product

If there is a finite set A and a finite set B, then there are $|A| \times |B|$ pairs of objects, the first from A and the second from B

Consider for simplicity $|A| = 5$ and $|B| = 3$



$$5 + 5 + 5 = 3 \times 5 = 15$$