

# **Алгоритмы Прима и Крускала построения остовного связного дерева минимального веса**



**Гуляева  
Татьяна Викторовна**  
учитель информатики и  
математики МБОУ СОШ №9  
с УИОП г.Павлово

2014 год

# Содержание

- [Повторение основных понятий теории графов](#)
- [Понятие остовного связного дерева](#)
- [Понятие цикломатического числа](#)
- [Алгоритм Прима](#)
- [Алгоритм Крускала](#)
- [Вопросы и задания](#)



# Основные понятия теории графов

**По горизонтали:**

1. Графы являются абстрактными моделями реальных объектов и их связей. Они состоят из множества вершин (узлов) и множества ребер (связей) между ними. Графы используются в различных областях науки и техники для решения задач оптимизации, маршрутизации, анализа сетей и т.д.

2. Путь в графе — это последовательность различных вершин, соединенных ребрами. Длина пути — это количество ребер в нем. Самый короткий путь между двумя вершинами называется кратчайшим путем. Алгоритмы поиска кратчайшего пути (например, алгоритм Дейкстры) являются основными инструментами теории графов.

3. Цикл в графе — это замкнутый путь, начинающийся и заканчивающийся в одной и той же вершине. Циклы играют важную роль в анализе графов, особенно в задачах распознавания циклов и в алгоритмах поиска циклов.

4. Деревья — это графы, не содержащие циклов. Они являются простыми и важными типами графов. Деревья используются для моделирования иерархических структур, поиска путей и т.д. Алгоритмы поиска в деревьях (например, алгоритм поиска в ширину) являются основными инструментами теории графов.

5. Вершинный граф — это граф, в котором вершины являются объектами, а ребра — отношениями между ними. Вершинные графы используются для моделирования социальных сетей, биологических систем и т.д.

6. Реберный граф — это граф, в котором ребра являются объектами, а вершины — отношениями между ними. Реберные графы используются для моделирования сетей, например, электрических сетей или сетей связи.

7. Полный граф — это граф, в котором каждая вершина соединена с каждой другой вершиной. Полный граф является одним из самых простых и важных типов графов. Он используется для моделирования ситуаций, в которых все объекты связаны друг с другом.

8. Графы являются абстрактными моделями реальных объектов и их связей. Они состоят из множества вершин (узлов) и множества ребер (связей) между ними. Графы используются в различных областях науки и техники для решения задач оптимизации, маршрутизации, анализа сетей и т.д.

9. Путь в графе — это последовательность различных вершин, соединенных ребрами. Длина пути — это количество ребер в нем. Самый короткий путь между двумя вершинами называется кратчайшим путем. Алгоритмы поиска кратчайшего пути (например, алгоритм Дейкстры) являются основными инструментами теории графов.

10. Цикл в графе — это замкнутый путь, начинающийся и заканчивающийся в одной и той же вершине. Циклы играют важную роль в анализе графов, особенно в задачах распознавания циклов и в алгоритмах поиска циклов.

11. Деревья — это графы, не содержащие циклов. Они являются простыми и важными типами графов. Деревья используются для моделирования иерархических структур, поиска путей и т.д. Алгоритмы поиска в деревьях (например, алгоритм поиска в ширину) являются основными инструментами теории графов.

12. Вершинный граф — это граф, в котором вершины являются объектами, а ребра — отношениями между ними. Вершинные графы используются для моделирования социальных сетей, биологических систем и т.д.

13. Реберный граф — это граф, в котором ребра являются объектами, а вершины — отношениями между ними. Реберные графы используются для моделирования сетей, например, электрических сетей или сетей связи.

14. Полный граф — это граф, в котором каждая вершина соединена с каждой другой вершиной. Полный граф является одним из самых простых и важных типов графов. Он используется для моделирования ситуаций, в которых все объекты связаны друг с другом.

вертикали



# Основные понятия теории графов

**По вертикали:**

8. Верифицируем неравноудаленными вершинами графа при перемещении.

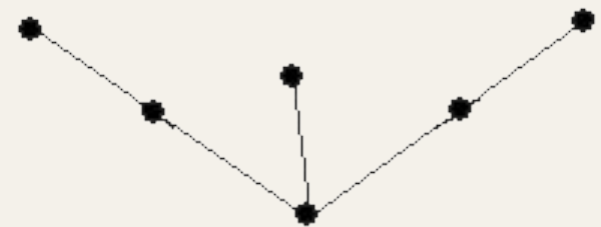
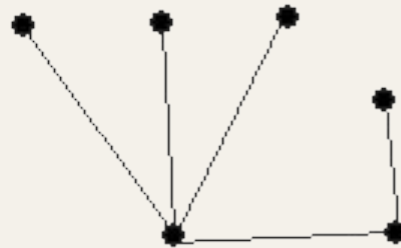
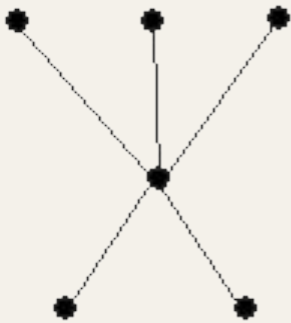
Перейдем к изучению  
НОВЫХ ПОНЯТИЙ



# Основные понятия теории графов

## Остовное связное дерево

**Остовной связный подграф** – подграф графа  $G$ , который содержит все его вершины и каждая вершина достижима из любой другой.



**Остовное связное дерево** – подграф, включающий вершины исходного графа  $G$ , не содержащего циклы, каждая вершина которого достижима из любой другой.



# Основные понятия теории графов

## Цикломатическое число

**Цикломатическое число**  $\gamma$  показывает, сколько ребер нужно удалить из графа, чтобы в нем не осталось циклов

$$\gamma = m - n + 1,$$

$m$  - количество ребер

$n$  - количество вершин



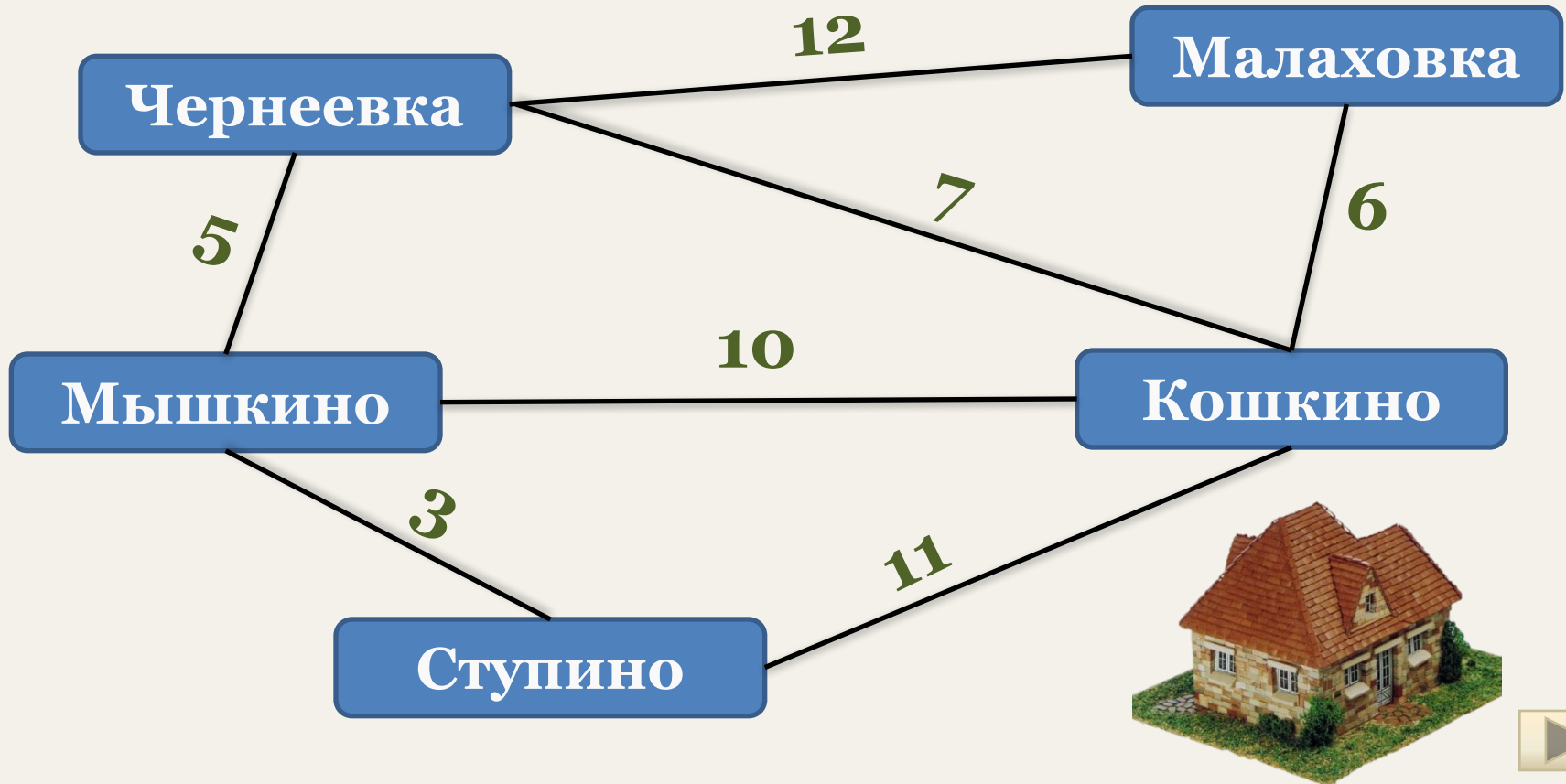
# Задача 1

В некотором районе было решено провести газопровод между пятью деревнями. От Кошкино до Мышкино идти 10 км, от Мышкино до Ступино – 3 км, от Кошкино до Малаховки – 6 км, от Малаховки до Черняевки – 12 км, от Кошкино до Ступино – 11 км, от Мышкино до Чернеевки – 5 км, от Кошкино до Чернеевки – 7 км. Как необходимо провести трубу, чтобы она соединяла все пять деревень, и затраты при этом были минимальными?



# Задача 1

Построим граф, моделирующий дороги, соединяющие деревни.





# Задача 1

Задача сводится к построению остовного связного дерева минимального веса.

Рассчитаем цикломатическое число.

**m** (количество ребер) равно **7**

**n** (количество вершин) равно **5**

$$\gamma = 7 - 5 + 1 = 3$$

Т.е. три деревни напрямую соединены газовой трубой не будут.



*(переходы по щелчку)*

# Алгоритм Прима

Пусть дан взвешенный неориентированный граф.

Для построения минимального остовного дерева необходимо:

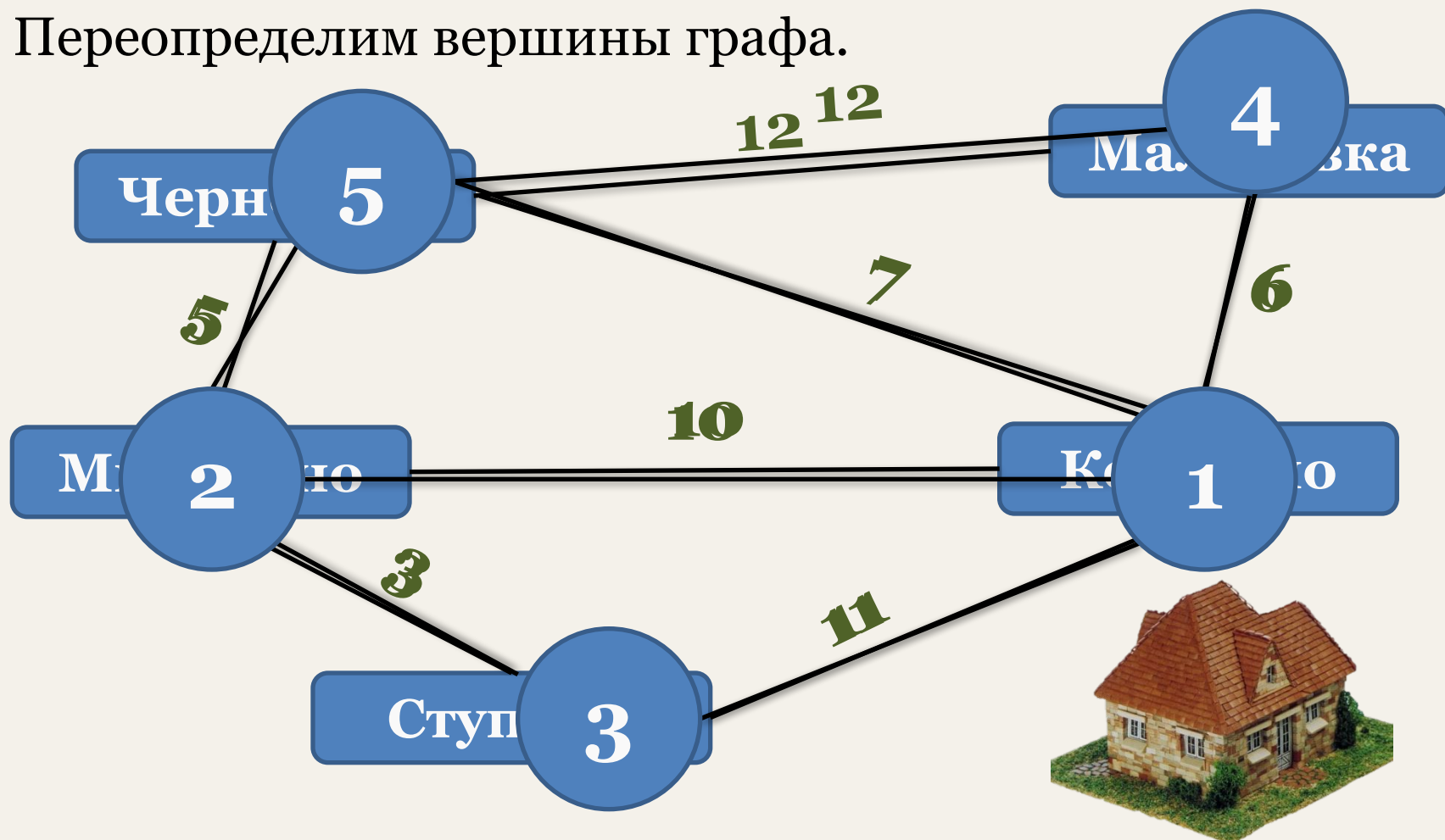
1. Представить граф в виде матрицы смежности
2. Найти в матрице наименьший элемент, соответствующий ребру, соединяющему  $i$ -ю и  $j$ -ю вершины графа
3. Вычеркнуть элементы  $i$ -й и  $j$ -й строки матрицы
4. Пометить  $i$ -й и  $j$ -й столбцы матрицы
5. В помеченных столбцах  $i$  и  $j$  найти наименьший элемент, отличный от уже найденного
6. Повторять пункты 3-5 до тех пор, пока не будут задействованы все вершины графа



*(переходы по щелчку)*

# Задача 1

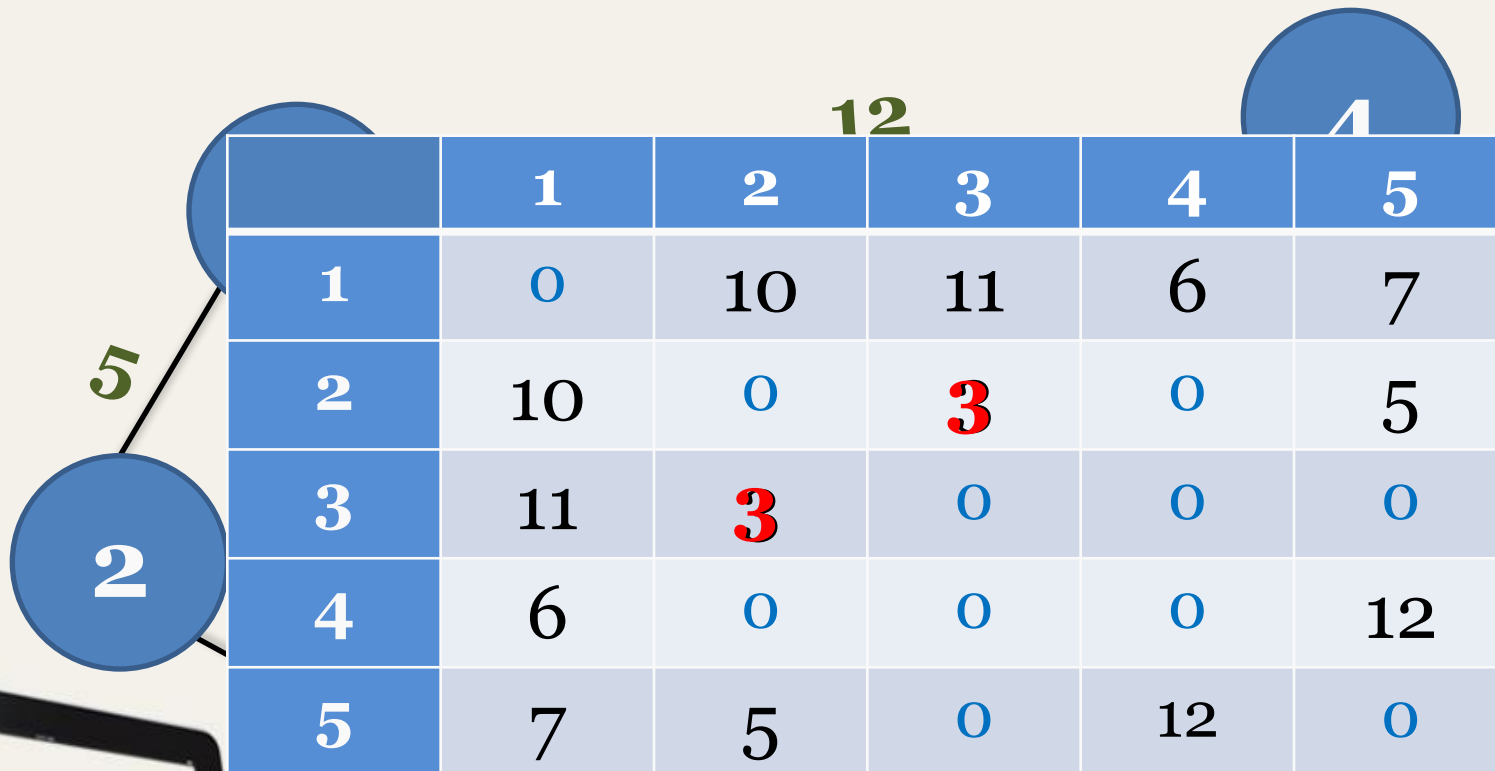
Решим задачу по алгоритму Прима.  
Переопределим вершины графа.



(переходы по щелчку)

# Задача 1

Представим граф в виде матрицы смежности.



The diagram shows a 5x5 adjacency matrix with nodes 1-5. Edges are highlighted with circles and weights: edge (1,2) with weight 5, edge (2,3) with weight 12, and edge (3,4) with weight 4. The matrix values are: (1,1)=0, (1,2)=10, (1,3)=11, (1,4)=6, (1,5)=7; (2,1)=10, (2,2)=0, (2,3)=3, (2,4)=0, (2,5)=5; (3,1)=11, (3,2)=3, (3,3)=0, (3,4)=0, (3,5)=0; (4,1)=6, (4,2)=0, (4,3)=0, (4,4)=0, (4,5)=12; (5,1)=7, (5,2)=5, (5,3)=0, (5,4)=12, (5,5)=0.

	1	2	3	4	5
1	0	10	11	6	7
2	10	0	3	0	5
3	11	3	0	0	0
4	6	0	0	0	12
5	7	5	0	12	0

Найдем минимальный элемент.

Он равен **3**



*(переходы по щелчку)*

# Задача 1

Вычеркнем 2-ю и 3-ю строки таблицы. А столбцы 2 и 3 выделим.

	1	2	3	4	5
1	0	10	11	6	7
2			<b>3</b>		
3					
4	6	0	0	0	12
5	7	<b>5</b>	0	12	0

Найдем минимальный элемент в выделенных столбцах. Он равен **5**



*(переходы по щелчку)*

# Задача 1

Вычеркнем 5-ю строку таблицы. А столбец 5 выделим.

	1	2	3	4	5
1	0	10	11	6	7
2			3		
3					
4	6	0	0	0	12
5		5			

Найдем минимальный элемент в выделенных столбцах. Он равен 7



*(переходы по щелчку)*

# Задача 1

Вычеркнем 1-ю строку таблицы. А столбец 1 выделим.

	1	2	3	4	5
1					7
2			3		
3					
4	6	0	0	0	12
5		5			

Найдем минимальный элемент в выделенных столбцах. Он равен **6**



*(переходы по щелчку)*

# Задача 1

Вычеркнем 4-ю строку таблицы. А столбец 4 выделим.

	1	2	3	4	5
1					7
2			3		
3					
4	6				
5		5			

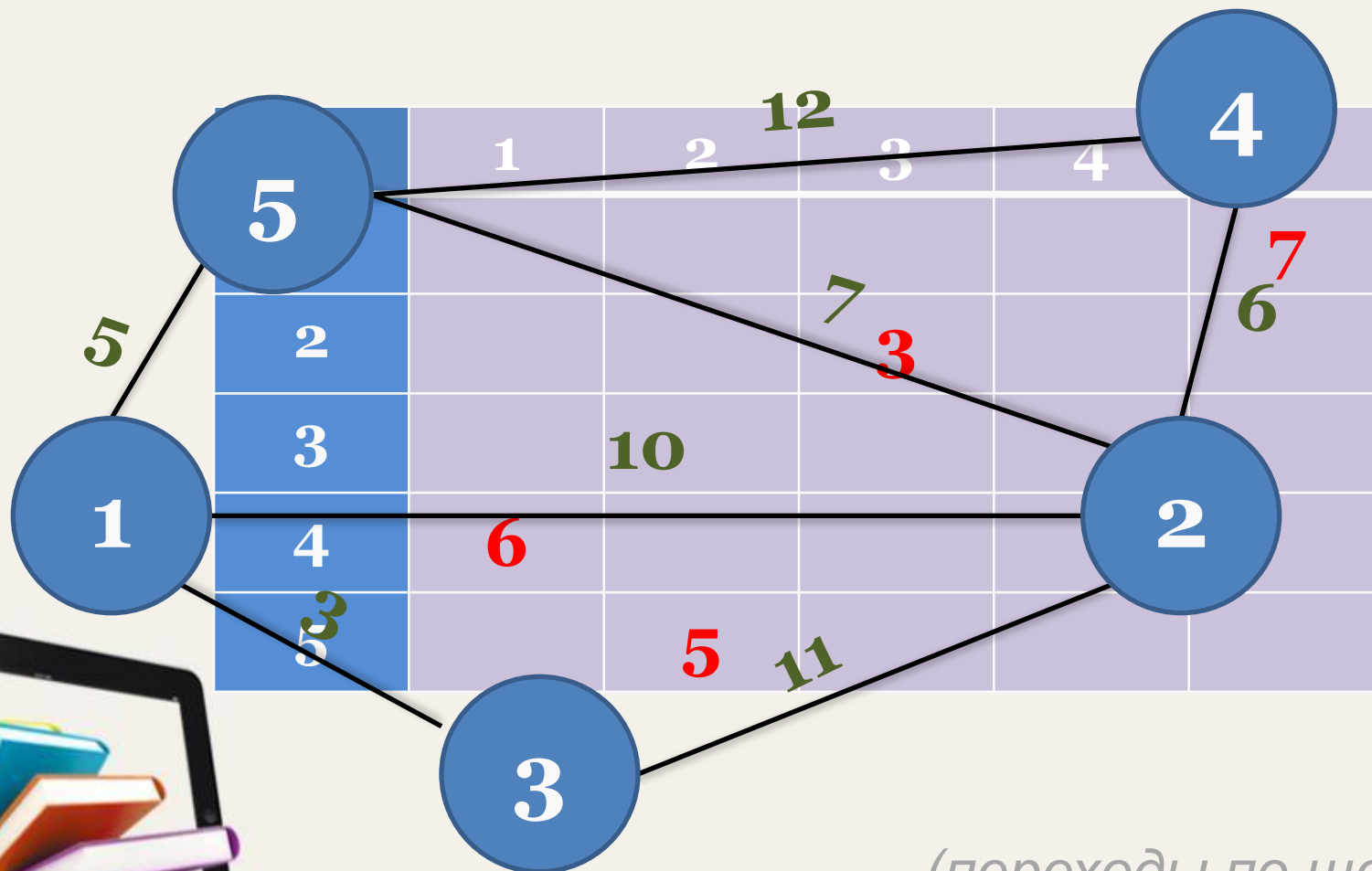


*(переходы по щелчку)*



# Задача 1

Получаем связное остовное дерево минимального веса.

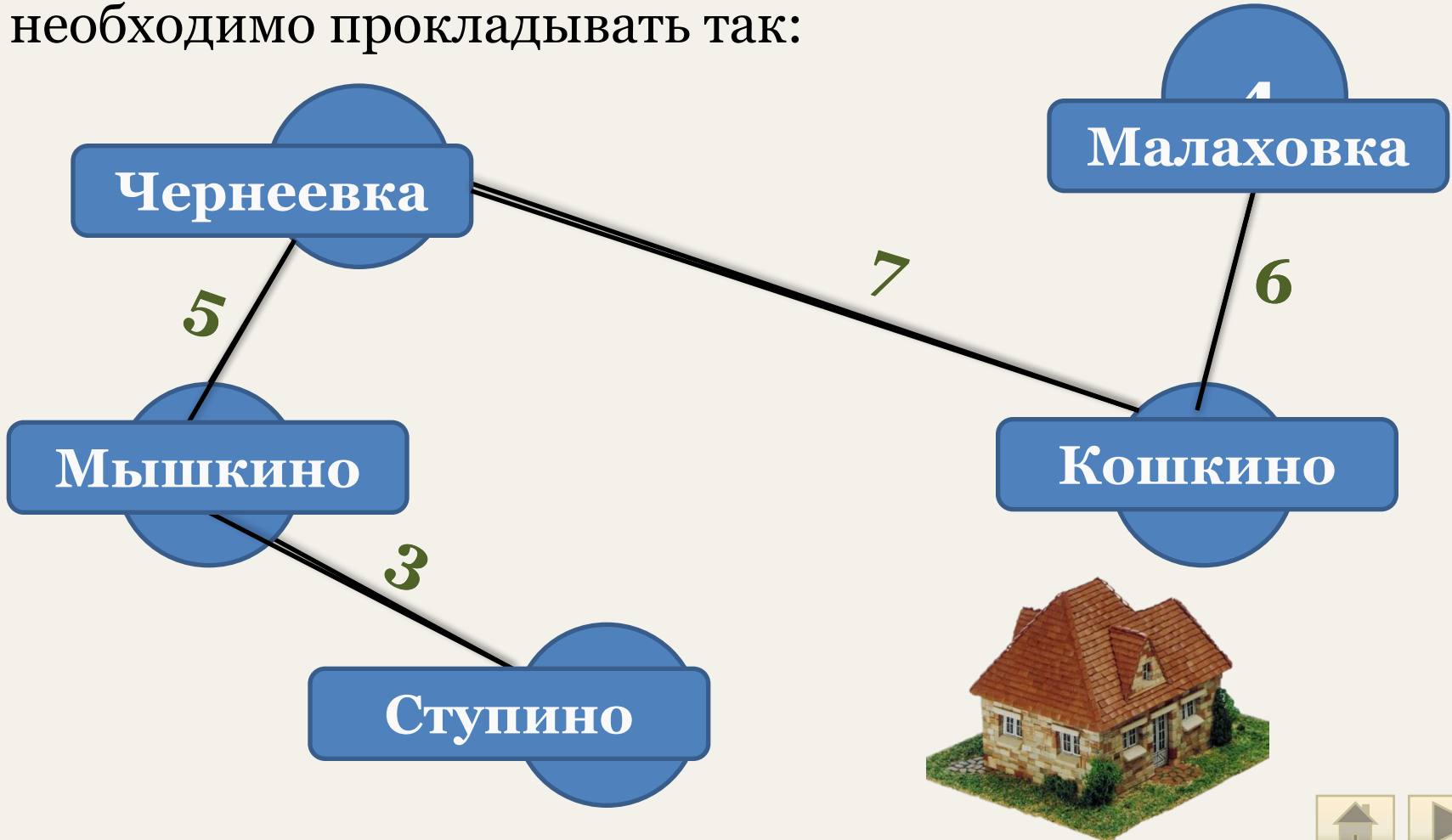


(переходы по щелчку)



# Задача 1

**Ответ:** газопровод с минимальными затратами необходимо прокладывать так:

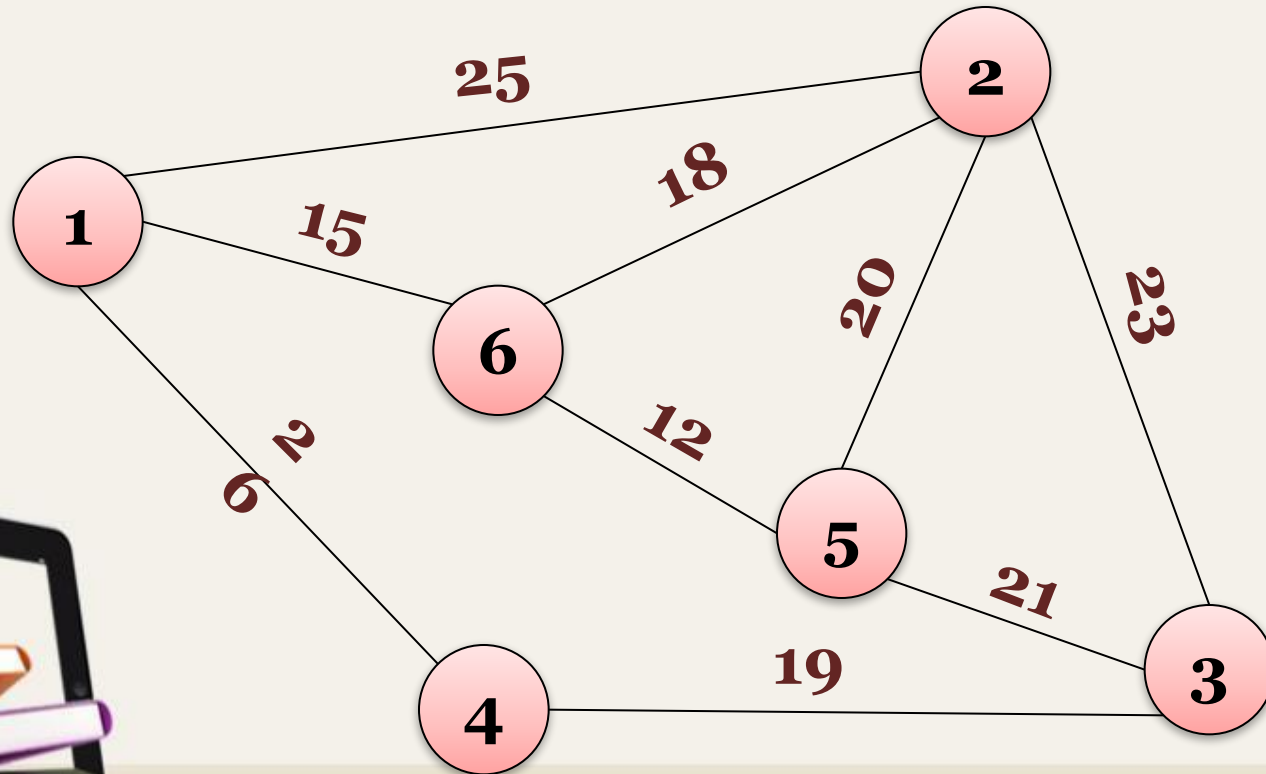


Протяженность газопровода – **21 км.**



# Задача 2

Даны города, часть которых соединена между собой дорогами. Необходимо проложить туристический маршрут минимальной длины, проходящий через все города.



# Задача 2

Задача сводится к построению остовного связного дерева минимального веса.

Рассчитаем цикломатическое число.

**m** (количество ребер) равно **9**

**n** (количество вершин) равно **6**

$$\gamma = 9 - 6 + 1 = 4$$

Т.е. четыре дороги, соединяющие города, не будут включены в туристический маршрут.

*(переходы по щелчку)*

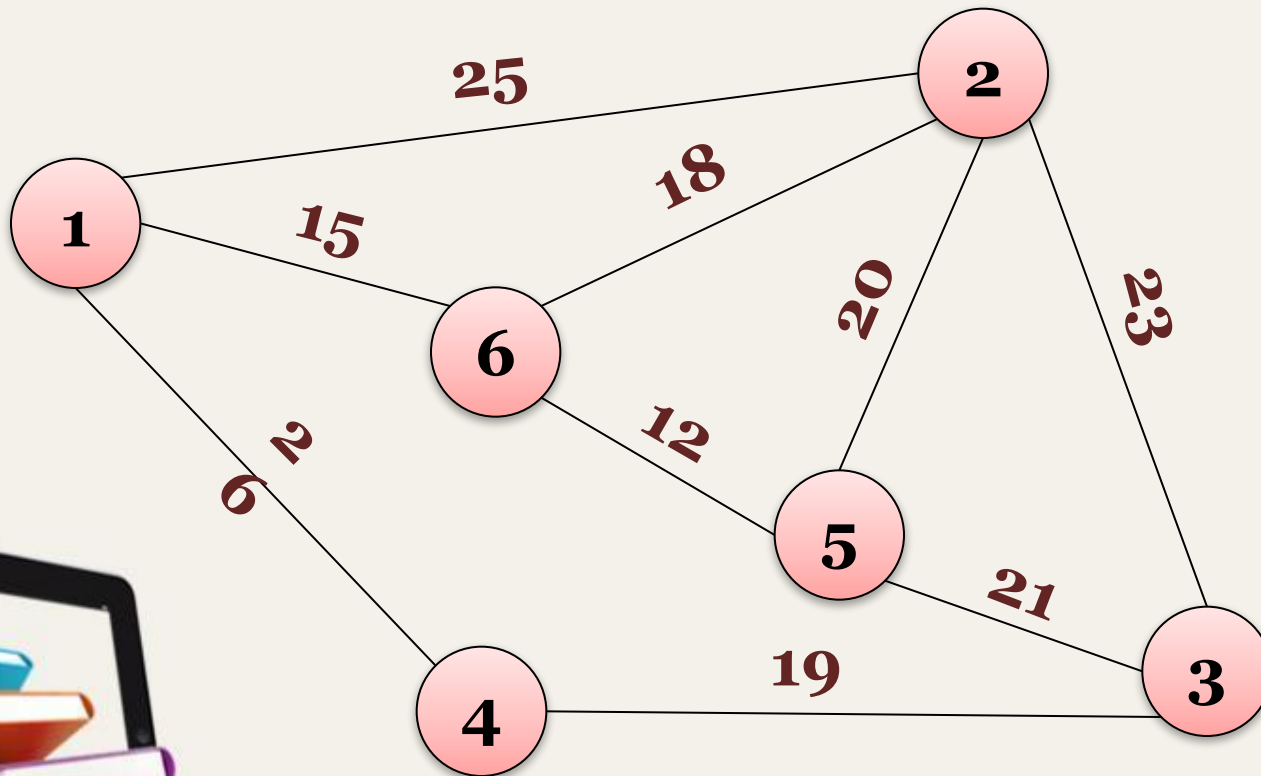
# Алгоритм Крускала

1. Удалить все ребра и получить остовный подграф с изолированными вершинами.
2. Отсортировать ребра по возрастанию.
3. Ребра последовательно, по возрастанию их весов, включаются в остовное дерево. Возможны случаи:
  - а) обе вершины включаемого ребра принадлежат одноэлементным подмножествам, тогда они объединяются в новое, связное подмножество;
  - б) одна из вершин принадлежит связному подмножеству, другая нет, тогда включаем вторую в подмножество, которому принадлежит первая;
  - в) обе вершины принадлежат разным связным подмножествам, тогда объединяем подмножества;
  - г) обе вершины принадлежат одному связному подмножеству, тогда исключаем данное ребро.
4. Алгоритм завершается, когда все вершины будут объединены в одно множество.



# Задача 2

Для определения туристического маршрута минимальной длины воспользуемся алгоритмом Крускала.

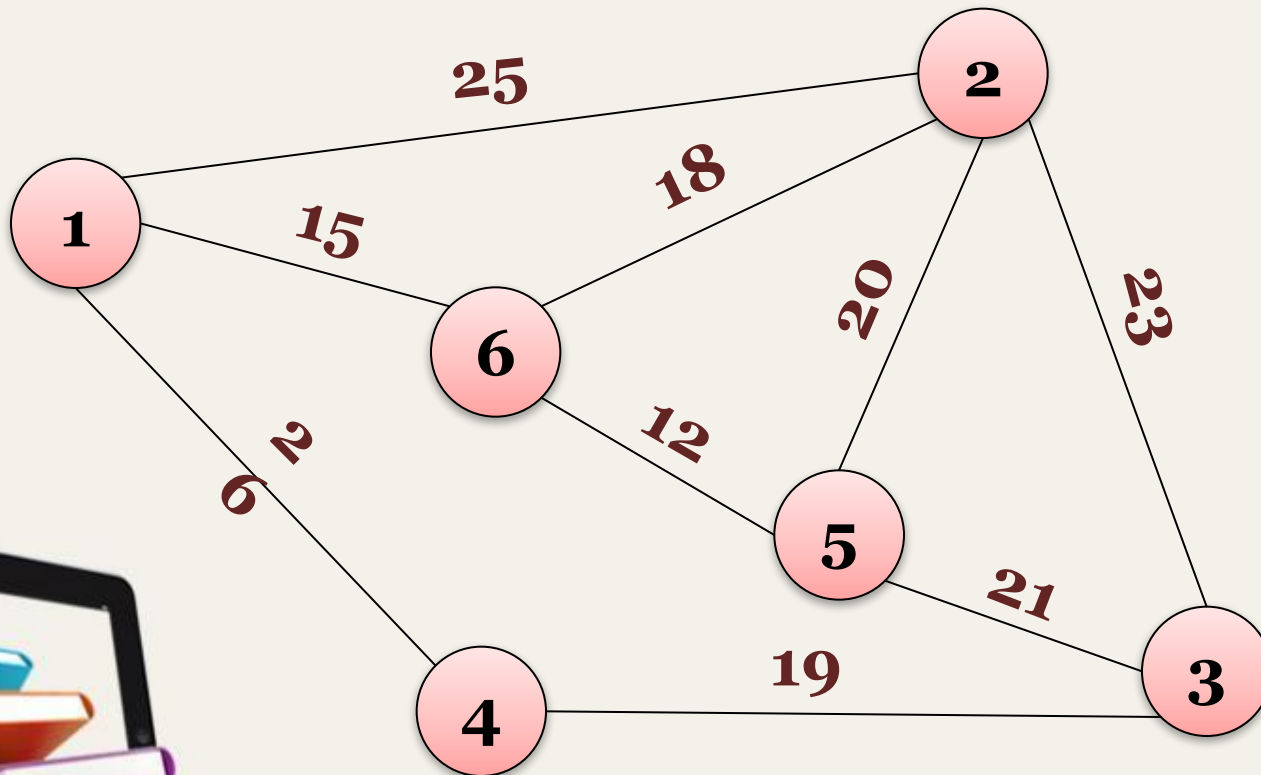


# Шаг 1

# Задача 2

Построим остовной подграф, содержащий только изолированные вершины.

Получаем шесть одноэлементных подмножеств.



пуск

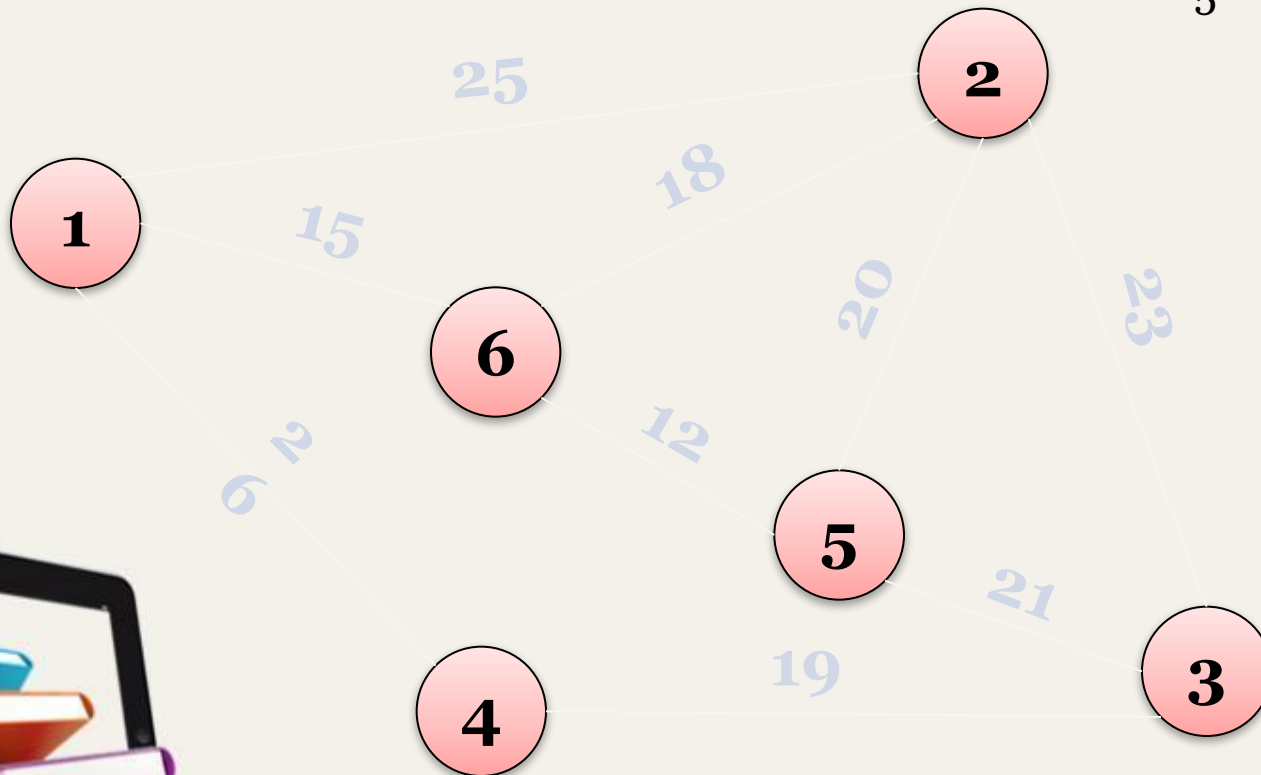


## Шаг 2

## Задача 2

Найдем ребро минимального веса и добавим его в остовной подграф.

Образуются связное подмножество вершин  $\{V_5, V_6\}$ .



пуск



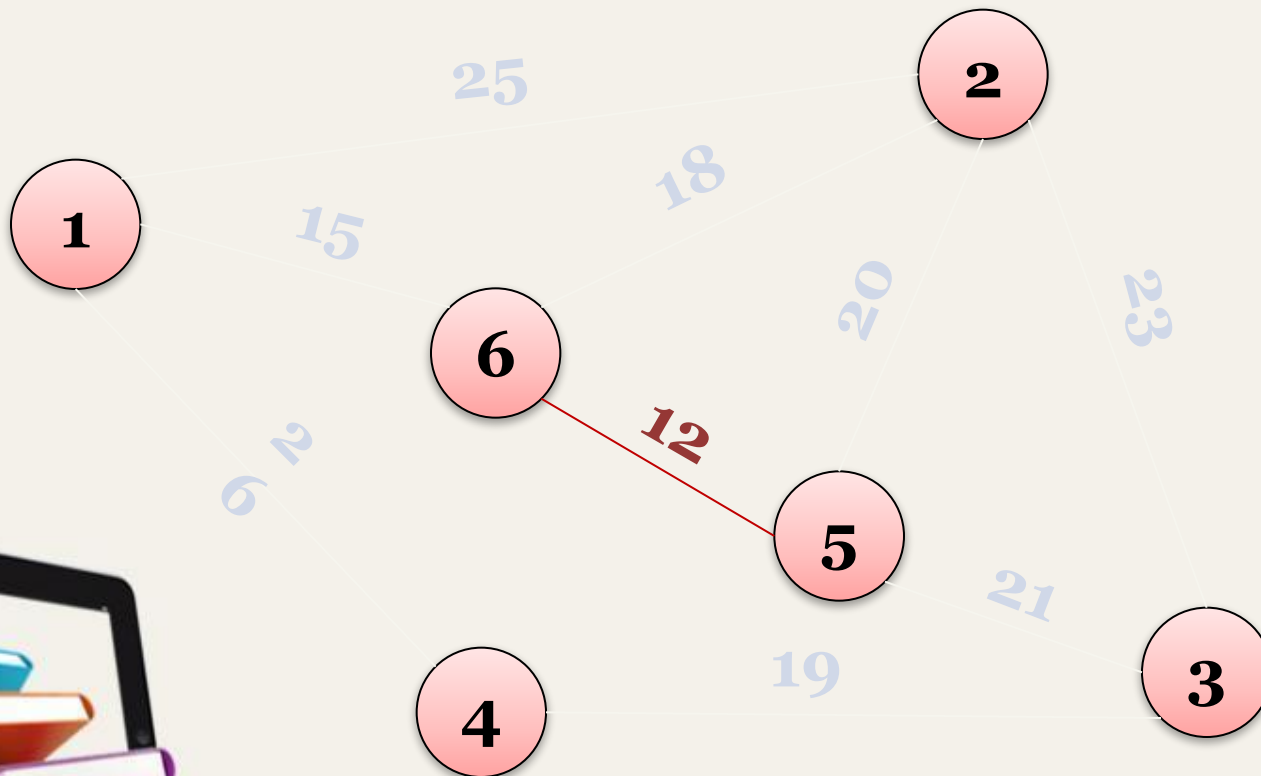


## Шаг 3

## Задача 2

Среди оставшихся ребер найдем ребро минимального веса и добавим его в остовной подграф.

Добавляем в подмножество вершин еще одну  $\{V_5, V_6, V_1\}$ .



пуск

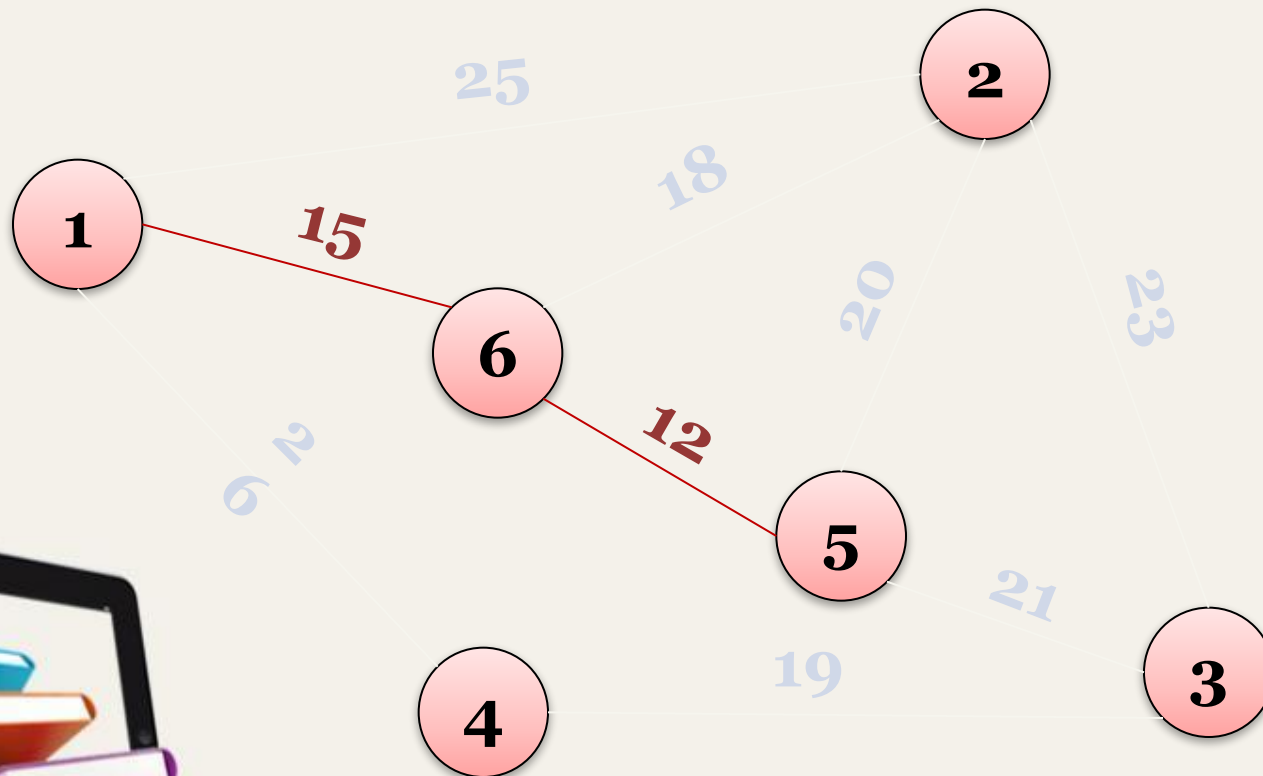


## Шаг 4

## Задача 2

Среди оставшихся ребер найдем ребро минимального веса и добавим его в остовный подграф.

Добавляем в подмножество вершин еще одну  $\{V_5, V_6, V_1, V_2\}$ .



пуск

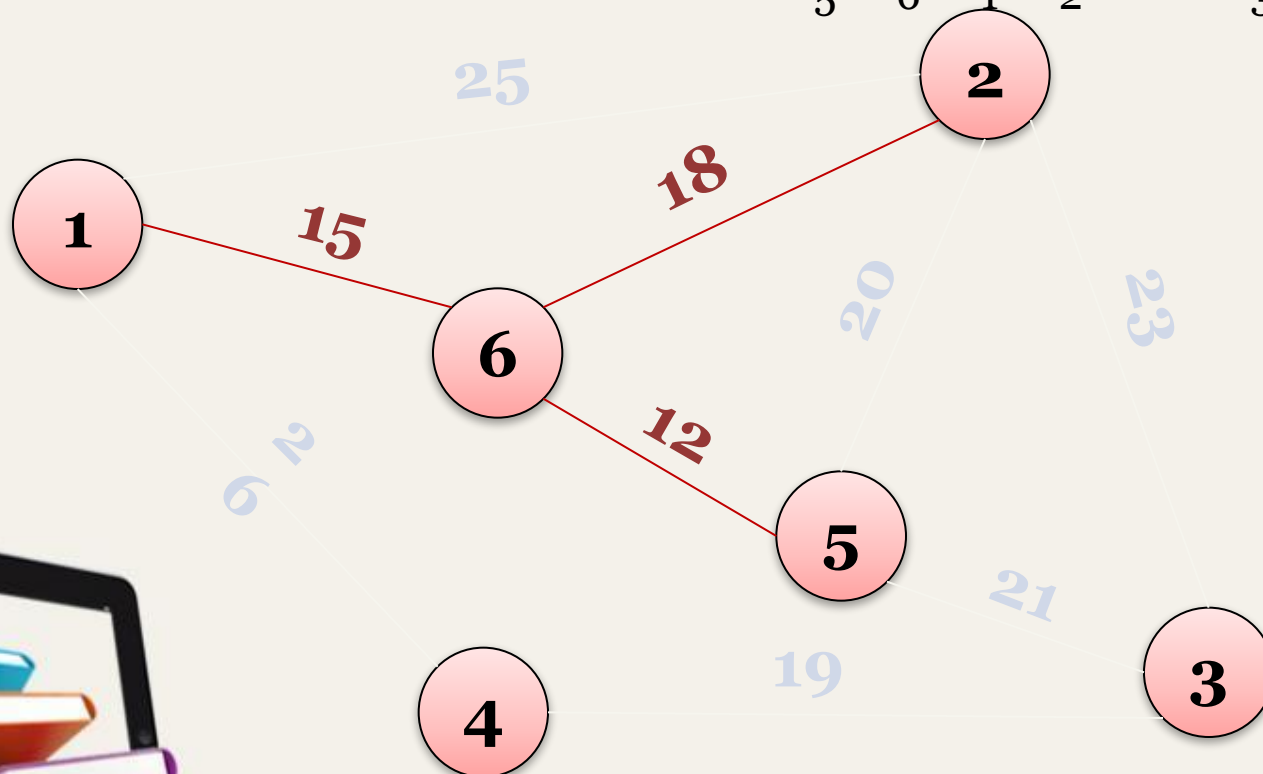


## Шаг 5

## Задача 2

Среди оставшихся ребер найдем ребро минимального веса и добавим его в остовный подграф.

Образуются два подмножества  $\{V_5, V_6, V_1, V_2\}$  и  $\{V_3, V_4\}$ .



ПУСК

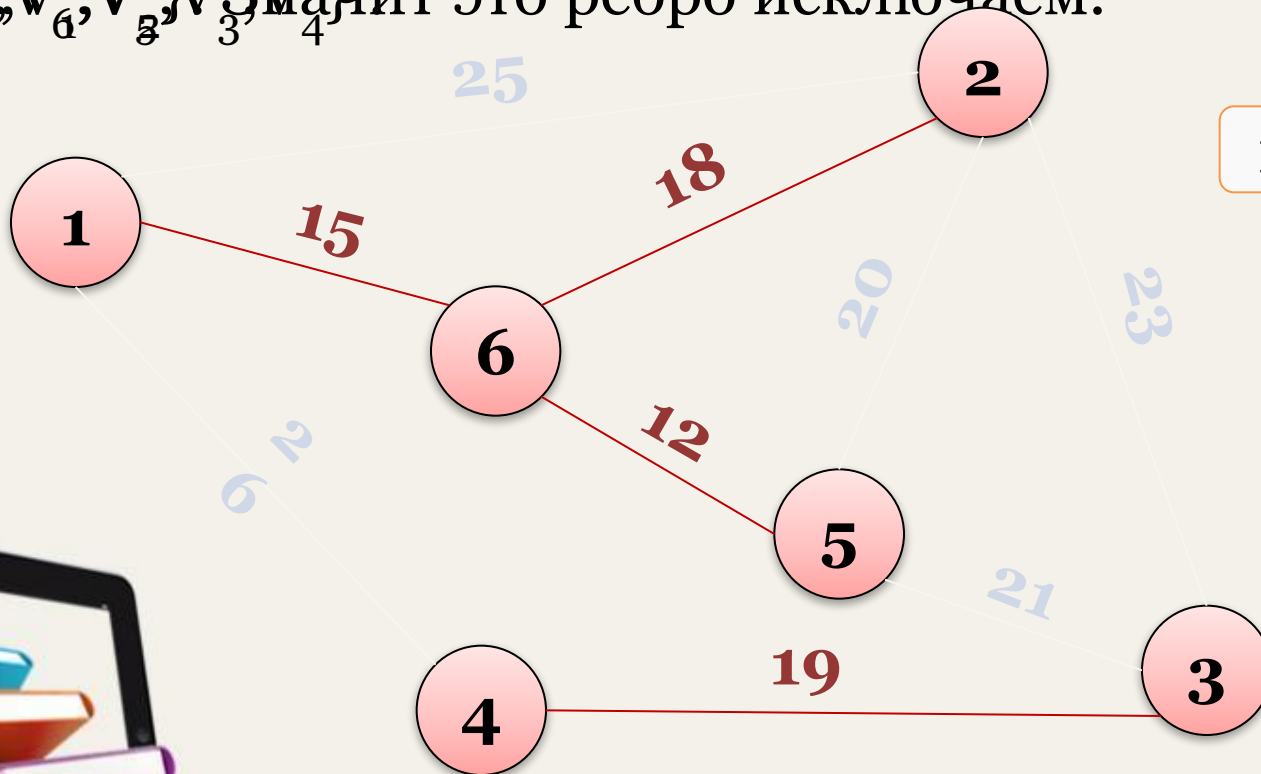


# Шаг 6

# Задача 2

Среди оставшихся ребер найдем ребро минимального веса и добавим его в остовный подграф.

Подобное ребро объединяет два компонента связности  $\{V_1, V_2, V_3, V_4, V_5, V_6\}$ . Значит это ребро исключаем.



пуск (2)

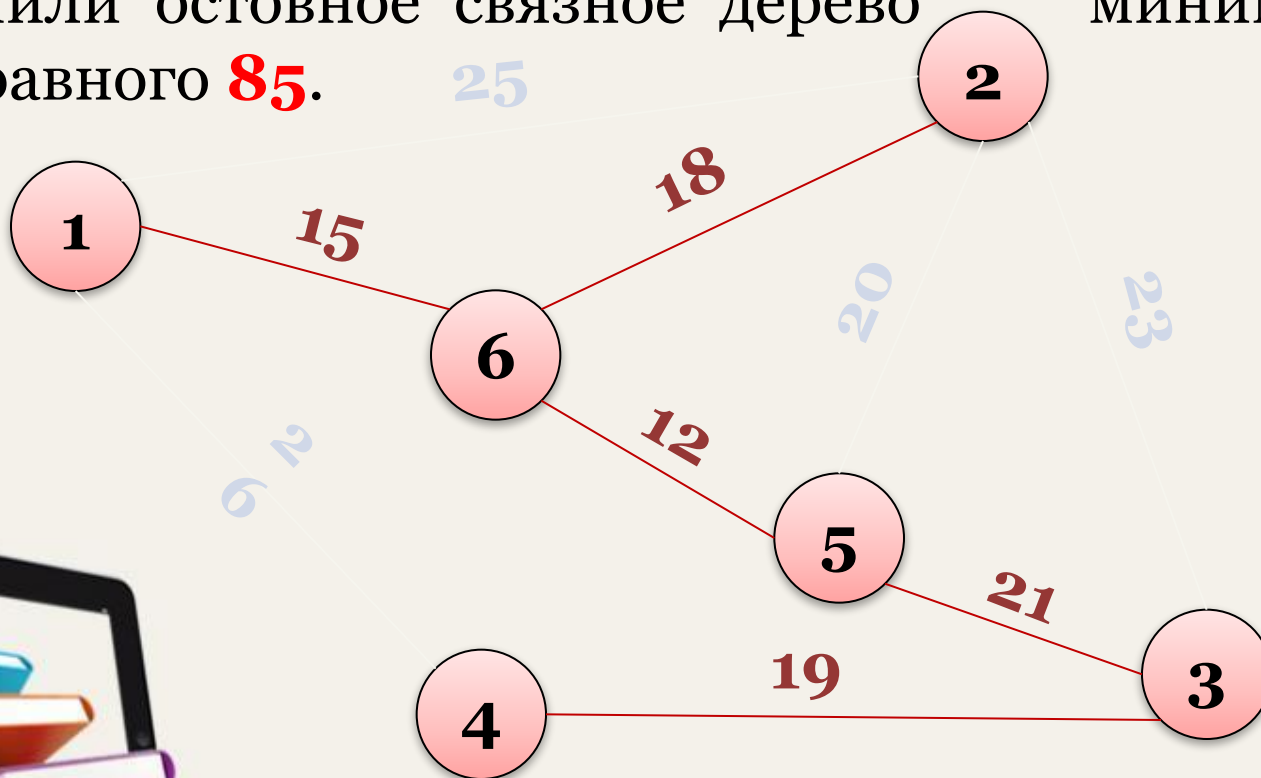


# Итог

## Задача 2

Остальные ребра включать в граф не надо, т.к. все их вершины уже принадлежат одному связному множеству.

Получили остовное связное дерево минимального веса, равного **85**.



# Вопросы

Построенный граф (в задачах 1 и 2) является

- **ОСТОВНЫМ**

В граф включены все вершины

- **СВЯЗНЫМ**

Все вершины в графе можно соединить маршрутами

- **ДЕРЕВОМ**

В графе отсутствуют циклы

- **С МИНИМАЛЬНЫМ ВЕСОМ**

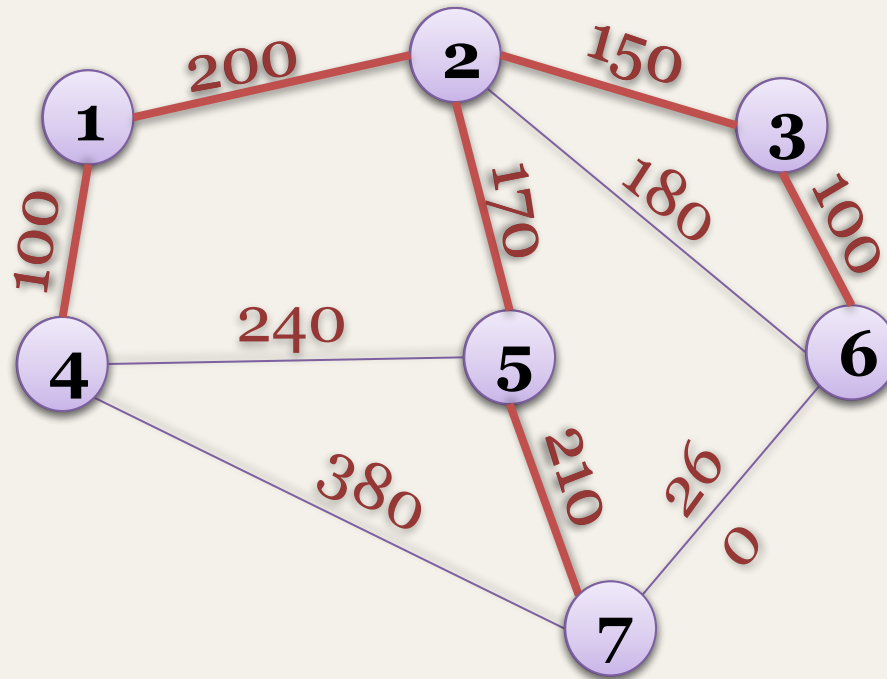
В граф последовательно включались ребра, отсортированные по возрастанию весов

**Почему?**



# Задача 3

На строительном участке необходимо создать телефонную сеть, соединяющую все бытовки. Для того, чтобы телефонные линии не мешали строительству, их решили проводить вдоль дорог. Схема участка изображена на рисунке.



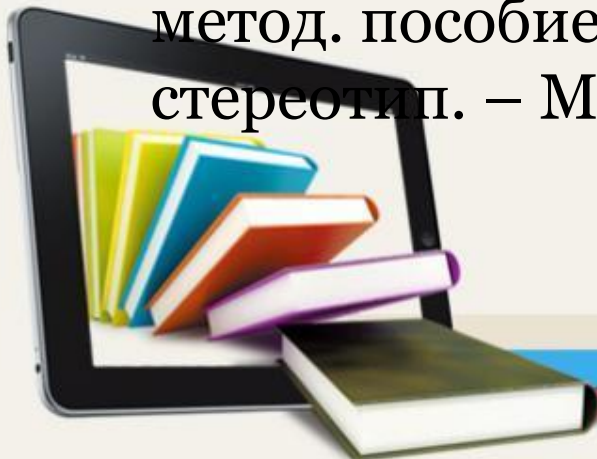
**Ответ**

Каким образом провести телефонные линии, чтобы их общая длина была минимальной? Общая длина телефонной линии равна **930** метров



# Источники

- Кроссворд создан на сайте и расположен по адресу <http://puzzlecup.com/?guess=3C2D4A01E0522AAU>
- Информатика и ИКТ. Профильный уровень: учебник для 11 класса / Н.Д.Угринович. – М.: Бином. Лаборатория знаний, 2010.
- Алгоритм Прима-Крускала (видео) [http://www.youtube.com/watch?v=vm\\_9-vnV7PE](http://www.youtube.com/watch?v=vm_9-vnV7PE)
- Занимательные задачи по теории графов: Учеб.-метод. пособие/ О.И.Мельников. – Изд-е 2-е, стереотип. – Мн.: «ТетраСистемс», 2001





# Источники изображений

- Изображение деревенского дома  
[http://www.diorama.com.ua/images/product\\_images/popup\\_images/2074\\_1.jpg](http://www.diorama.com.ua/images/product_images/popup_images/2074_1.jpg)
- Изображение связанных деревьев  
[http://xreferat.ru/image/54/1306491707\\_19.png](http://xreferat.ru/image/54/1306491707_19.png)



ВЫХОД