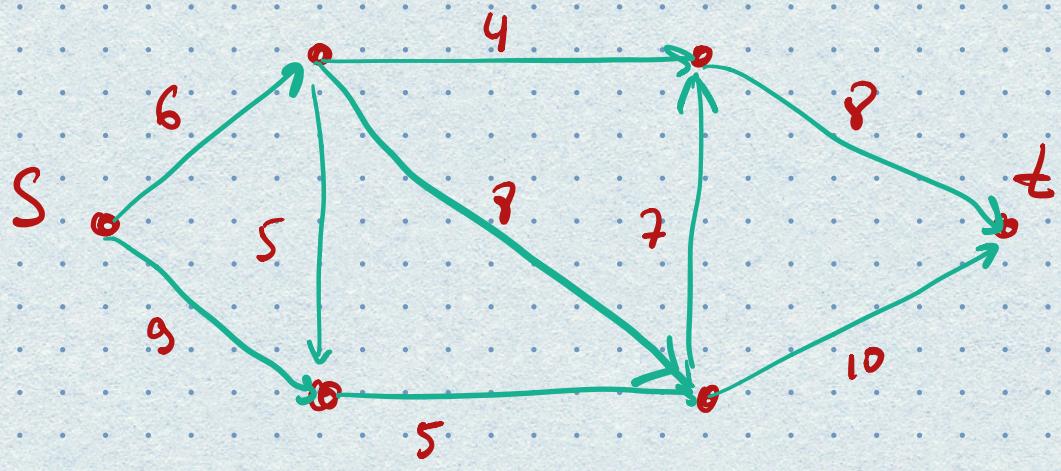


Алгоритм Форда - Фокеусона



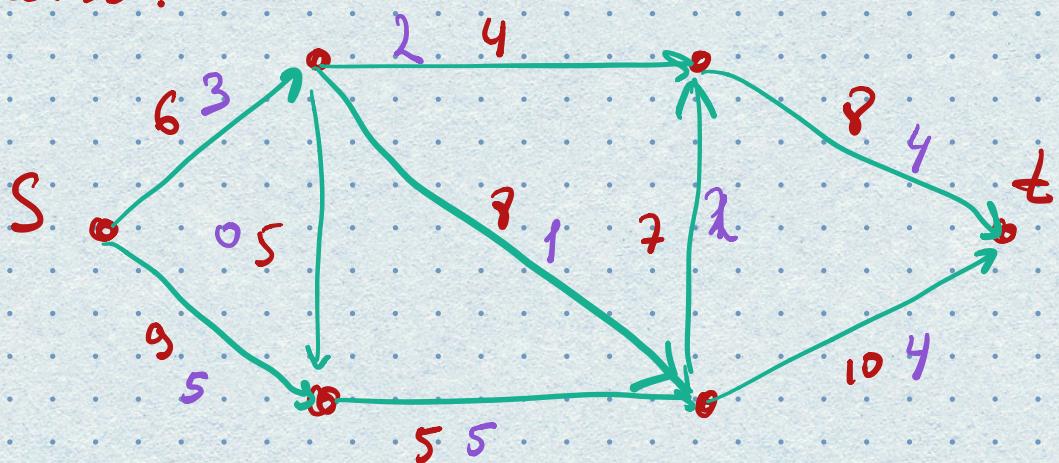
исток - S

сток - t

вес ребра - пропускная способность ребра.

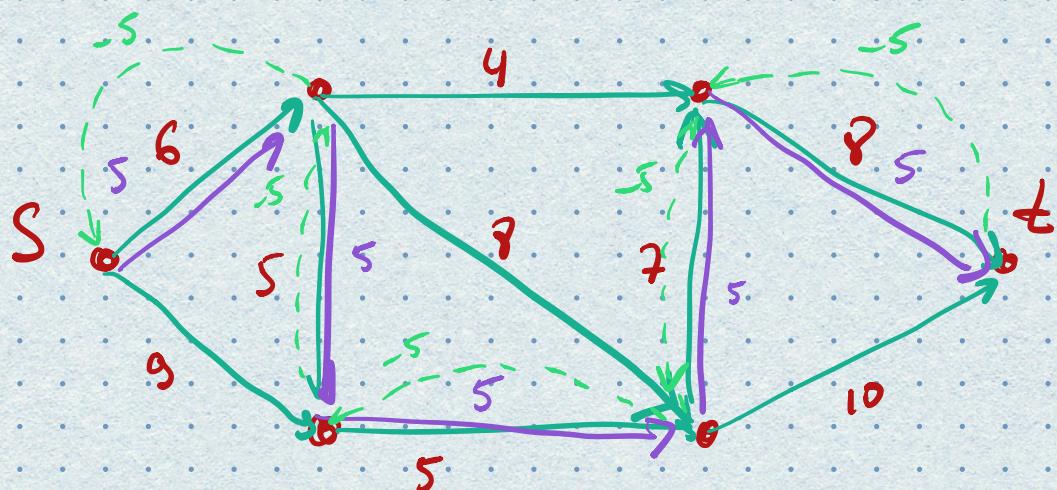
сумма по всем входящим ребрам в вершину =
сумма всех исходящих  (кроме S и t)

веса > 0 .

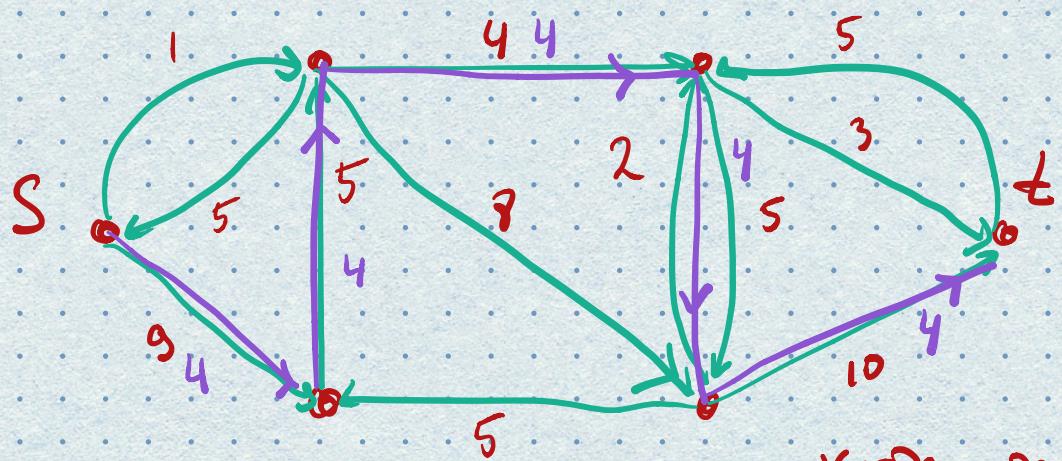
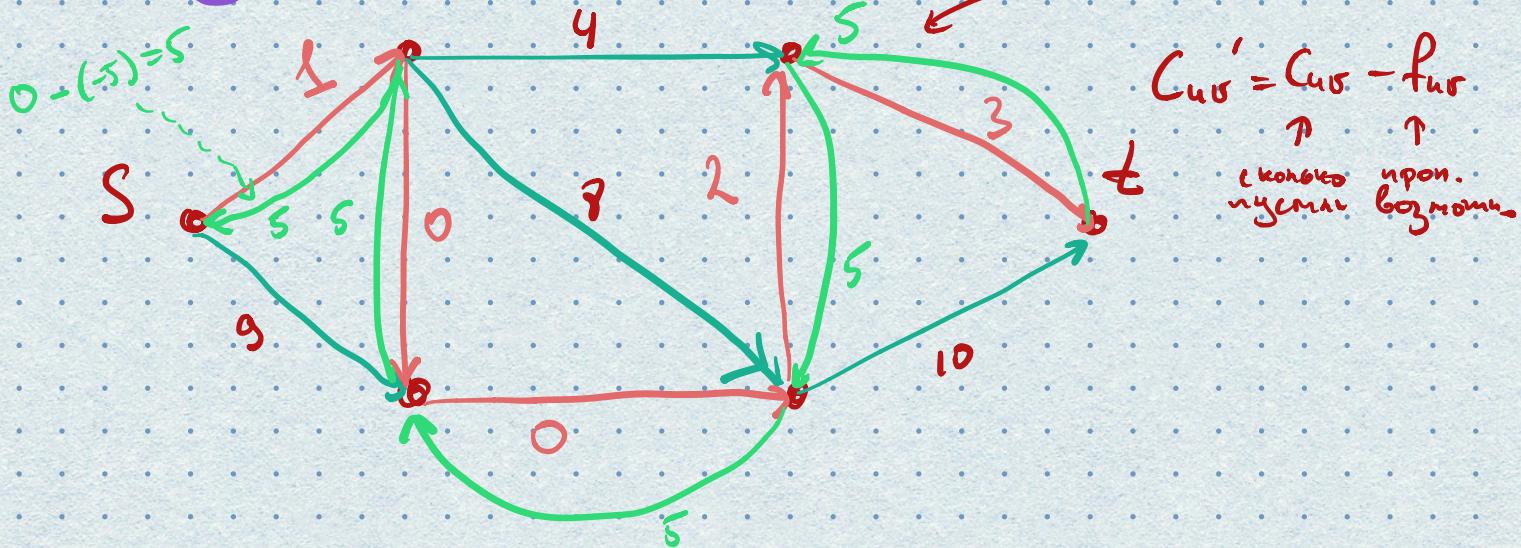


 - нисходящий поток

максимальный поток - хотим как можно больше перенаправить из S в t . В примере 8.

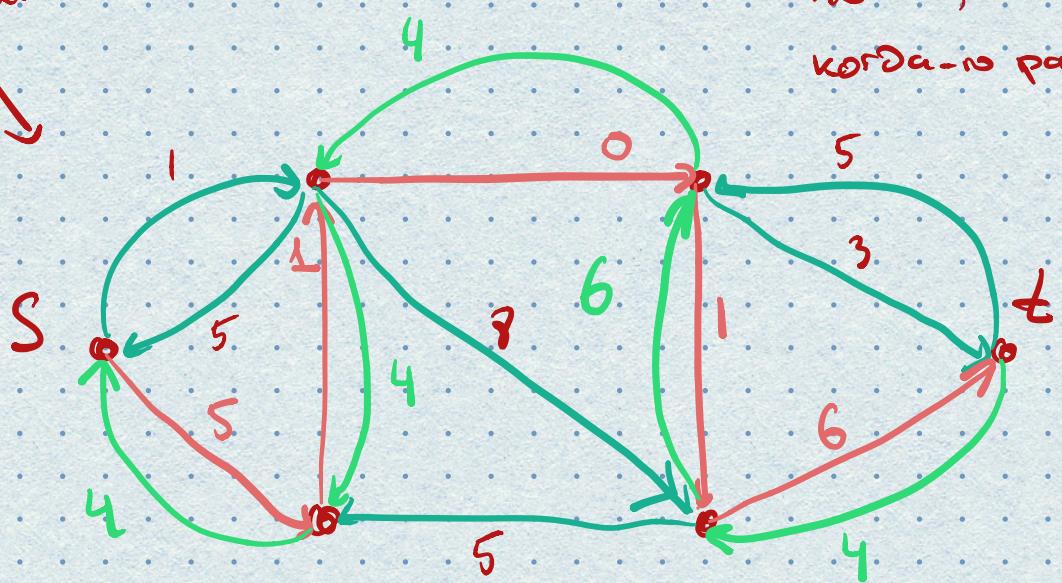


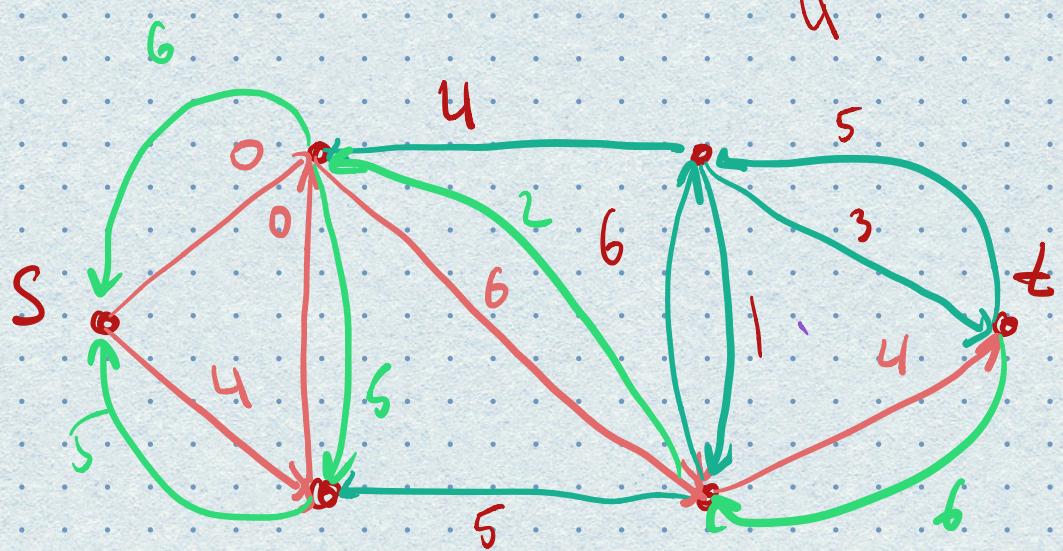
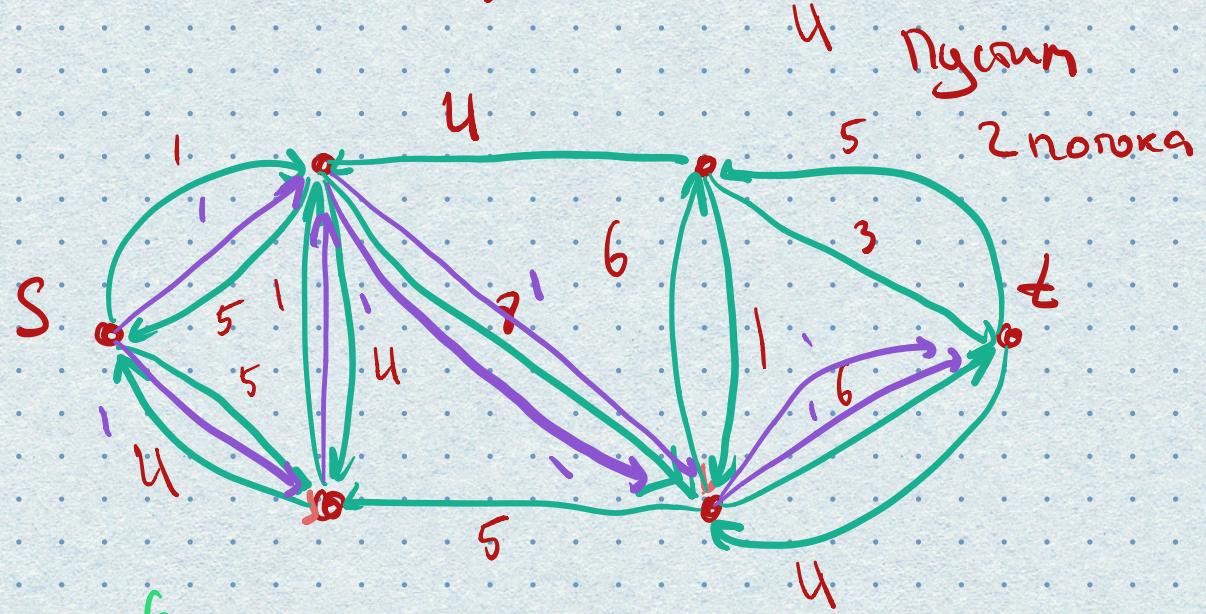
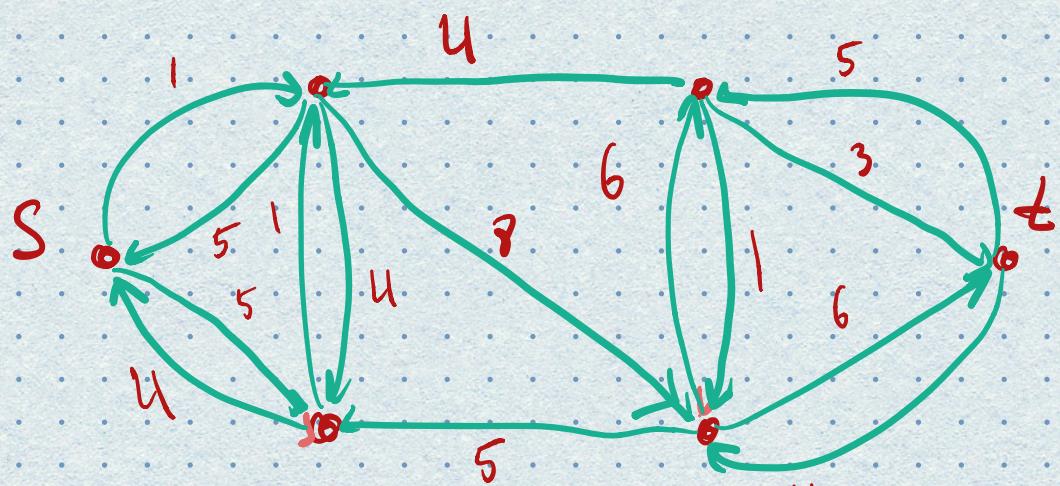
Будем какое-то подож.



Когда идем по обратному пути, то увеличиваем шагок, который нужно сделать для рабоче.

остановка сеза





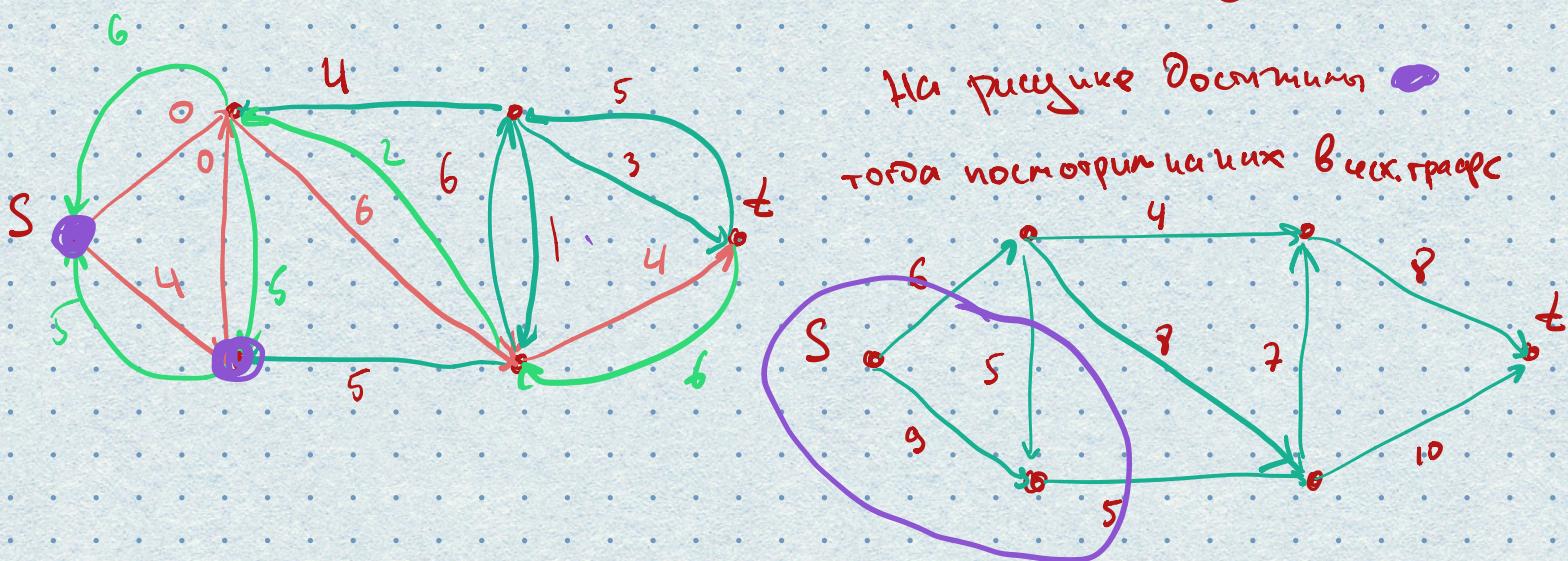
Больше пуским ногок не
можем \Rightarrow ногок макс

Min Cut

Хотим разделить вершины графа на 2 множества так, чтобы сумма всех выходящих рёбер была мин.

как найти:

1. ищем все вершины, достижимые из S . \rightarrow это искомое множество



$$\text{Max Flow} = 11 = \text{MinCut}$$

и (ход. рёбра): $5 + 6 = 11$

Алгоритм может не со = мсб

Сложность: $|F| \times |E|$ осталось будущее

$O(|V| |E|^2)$ - осталось вспоминать

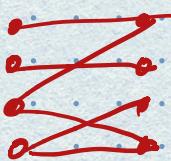
Алгоритм Эдмондса-Карпа

ищет не любой путь из S в t , а самый короткий.
т.е. искончлен δ в ширину.

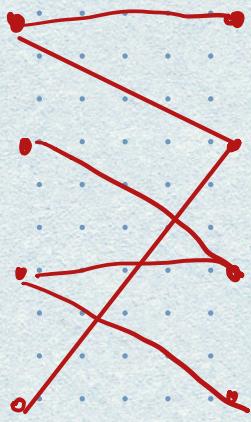
Алгоритм Куна

2 дополнительных графа - граф, у которого вершины можно разделить на 2
множества, и ребра будут идти из одного множества в

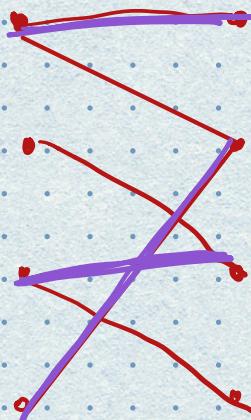
другое



Max паросочетание - наибольшее количество пар
каждая вершина входит только в одно ребро



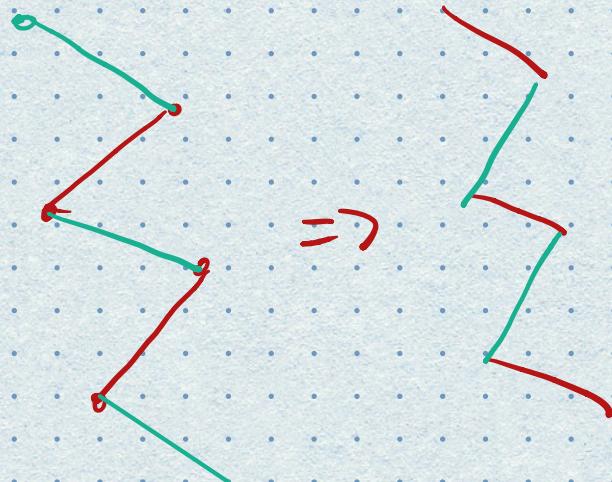
паросочетание:



увениц. степ:

m - не в пароср.

m в пароср.



момент и извергровав.

Т. Берна:

M_{\max} напосл \Leftrightarrow относительно M нет убен. пути ре-

шб: Если $x \in X$ не суз. убен. пути относительно M ,
то $x \in X$ нет и убен. пути относительно M' (M' находит $y \in Y$).

Алгоритм: шаги убен. путей пока момент.

Менем.