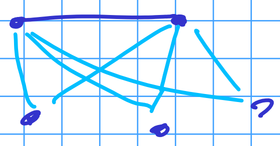
Кратчайшие пути от всех вершин

1. N раз Беллмана-Форда $O(V \cdot VE) = O(V^4)$
2. N раз Дейкстра $O(V \cdot (V+E)\log V)$



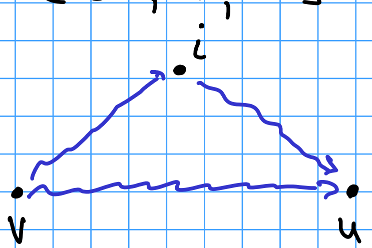
Алгоритм Флойда-Уоршалла

Динамическое программирование

1. $d[v, u, i]$ – кратчайший путь из v в u, проходящий по вершинам от $1 \ldots i$

   $d[v, u, 0]$ – матрица смежности
   $d[v, u, 1]$ – матрица путей из одного ребра или из двух рёбер, проход. ч/з вершину № 1.
   ......
   $d[v, u, n]$ – матрица кратчайших путей

2. $d[v, u, i] = \min\{d[v, u, i-1], d[v, i, i-1] + d[i, u, i-1]\}$



3. Порядок: по третьему индексу.

   Время: $O(V^3)$
   Память: $O(V^2)$

Floyd-Warshal

for $v, u \in [1..n]$: $d[v,u,0] = \infty$

for $(v,u) \in E$: $d[v,u,0] = w(u,v)$

for $v \in [1..n]$: $d[v,v,0] = 0$

for $i \in [1..n]$:

  for $v \in [1..n]$:

    for $u \in [1..n]$:

$$d[v,u,\cancel{i}] = \min\{d[v,u,\cancel{i-1}],\ d[v,i,\cancel{i-1}] + d[i,u,\cancel{i-1}]\}$$

NB: Достаточно таблички $n \times n$.

Вопрос: как восстановить пути?

Для $\forall$ пары вершин нам нужно хранить последнюю вершину, выбранную тут

Доп. $O(V^2)$ памяти

Алгоритм Джонсона

Лемма:

$\exists f: V \to \mathbb{R}$

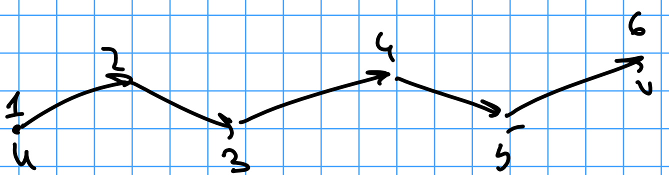
$\Rightarrow w'(v,u) = w(v,u) + f(v) - f(u)$

Утв: кратчайшие пути в графе с весами w' такие же, как в графе с весами w.

$\triangleright$ $d(v,u)$ — кратчайшее расстояние от v до u относит. w

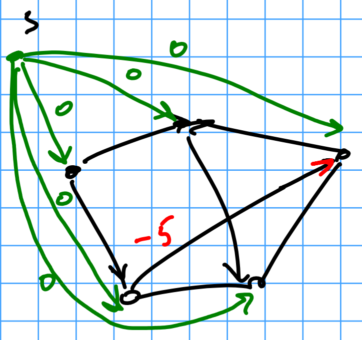
$d'(v,u)$ — " — относительно w'

$$d'(v,u) = w'(1,2) + w'(2,3) + \ldots + w'(5,6) =$$



$$= w(1,2) + f(1) - \cancel{f(2)} + w(2,3) + \cancel{f(2)} - \cancel{f(3)} + w(3,4) + \cancel{f(3)} - f(4) + \ldots =$$

$= \underline{d(v,u)} + \underline{f(v) - f(u)}$ ◁

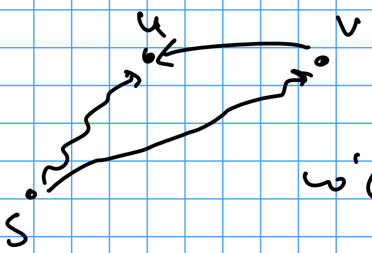


Запустим Беллмана-Форда от S и получим $d_S(v)$

Возьмем в качестве f из леммы ф-цию d_S.

Утв: $w'(v,u) = w(v,u) + \underline{d_S(v) - d_S(u)} \geq 0$?

▷ $d_S(u) \leq d_S(v) + w(v,u)$



$\underline{d_S(v) - d_S(u)} \geq \underline{-w(v,u)}$

$w'(v,u) = w(v,u) + \underline{d_S(v) - d_S(u)} \geq$

$\geq w(v,u) - w(v,u) \geq 0$ ◁

Теперь запустим N раз алг. Дейкстры.

↓

$d[v,u]$ кратчайшие пути м/у всеми парами вершин

↓ (нивелируем добавку)

$d^*[v,u] = d[v,u] - \underline{d_S(v) + d_S(u)}$

Когда это нужно?

$O(V^3)$ – Ф-У

E – ?

$O(V \cdot (V+E) \log V + VE) = o(V^3)$

$O(VE \log V) = o(V^3)$

$E = o(V^2 / \log V)$

Задача о редакционном расстоянии (Расстояние Левенштейна)

S U N N Y   Д О Ж Д Ь
S N O W Y   Д Р О Ж Ь

Три вида редакций:
1. замена
2. удаление
3. вставка

Задача о выравнивании

3
S U N N - Y
S - N O W Y

2
Д - О Ж Д Ь
Д Р О Ж - Ь

4
S U N N - - Y
S - N - O W Y

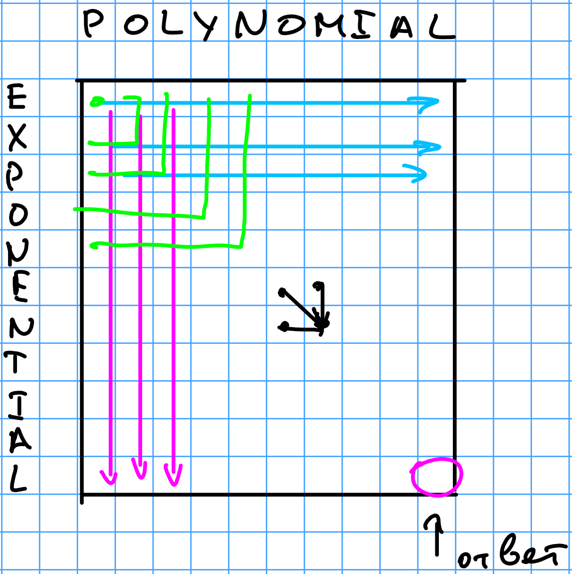
≡ Редакционное расстояние — min выравнивание

S_1 = POLYNOMIAL
S_2 = EXPONENTIAL

1. Мн-во подзадач $E(i, j)$ – min выравн. строк $S_1[1..i]$ и $S_2[1..j]$

$$E(i,j) = dist(S_1[1..i], S_2[1..j])$$
$$E(4,5) = dist(\text{"POLY"}, \text{"EXPON"})$$

2. $$E(i,j) = \min \begin{cases} E(i-1, j-1) + [S_1[i] \neq S_2[j]], & \text{замена} \\ E(i, j-1) + 1 & \text{вставка} \\ E(i-1, j) + 1 & \text{удаление} \end{cases}$$



		Д	О	Ж	Д	Ь
	0	1	2	3	4	5
Д	1	0	1	2	3	4
Р	2	1	1	2	3	4
О	3	2	1	2	3	4
Ж	4	3	2	1	2	3
Ь	5	4	3	2	2	2

ответ

↓ вставка

→ удаление

↘ замена

↘ та же буква.

$E(4,5) = \min$

1. замена.

POL	Y
EXPO	N

2. вставка

POLY	–
EXPO	N

3. удаление

POL	Y
EXPON	–

Время: $O(|s_1| \cdot |s_2|)$
Память: $O(|s_1| \cdot |s_2|)$

Вопрос:

Как решить за $O(|s_1| + |s_2|)$ памяти и то же время?