Bellman ford Algorithm

$s \to v$. at most i edges

P = shortest s-v path - (might have negative cycles)

cycles are permitted.

* case (i) P shortest
  such that $P \leq (i-1)$

* case (ii) P' takes to w is $(i-1)$ edges.
  such that $P' + (w, v)$ is optimal.

Bellman ford : Dynamic programming.

$\xrightarrow{\text{budget}}$

$\downarrow$ vertices.

budget forloop first. //

vertices forloop next. //

$\xrightarrow{\text{Budget}}$

$\downarrow$ vertices.

for every Budget

we need to go through

n d_v for every v.

$$\sum_{i}^{n} 1_i \times \sum_{j=1}^{n} d_{v_j}$$

$$= n \times m$$

$$\sum_{j=1}^{n} d_{v_j} = m$$

$= O(nm)$.

Dijkstra. $O(m \log n)$.