Bellman-Ford algorithm

input : $\mathbb{R}$-weighted graph $G = (V, E, \lambda)$, source vertex $s \in V$

output: if there is a negative weighted cycle reachable from s, then the algorithm prints out "negative weighted cycle"; and if there is no negative weighted cycle reachable from s, then it outputs the shortest-path weight $d_{s,v}$ for every $v \in V$

variable: $i \in \mathbb{N}$, $\delta : V \to \mathbb{R}_\infty$;

begin
  for every $u \in V$ **do** $\delta(u) := \delta^{in}(u)$;
  for $i := 1$ **to** $|V| - 1$ **do**
    for every $(u, v) \in E$ **do** $relax(u, v)$
  if for some edge $(u, v) \in E$ we have $\delta(v) > \delta(u) + \lambda(u, v)$
    then print("negative weighted cycle")
    else for every $v \in V$ **do** $d_{s,v} := \delta(v)$
end

where

$$\delta^{in}(u) = \begin{cases} \infty & \text{if } u \neq s \\ 0 & \text{otherwise} \end{cases}$$

and

$relax(u, v)$:
if $\delta(v) > \delta(u) + \lambda(u, v)$ **then** $\delta(v) := \delta(u) + \lambda(u, v)$.