Nejkratší cesta v grafu

VSTUP:
Graf $G = (V, E)$, vrchol $v_0 \in V$, délky hran $l(e)$ pro $e \in E$.

VÝSTUP:
Pro každý vrchol $v \in V$ délka $h(v)$ nejkratší cesty z vrcholu v_0 do vrcholu v.
Předposlední vrchol $P(v)$ na nejkratší cestě (slouží k rekonstrukci cesty).

PSEUDOKÓD:

1. For v in V
2. $h(v) \leftarrow +\infty$
3. $P(v) \leftarrow$ None
4. $h(v_0) \leftarrow 0$
5. otevřeme vrchol v_0 (tj. vložíme ho do datové struktury, která může být fronta, list, halda)
6. While existuje otevřený vrchol (tedy fronta, list či halda je neprázdná)
7.   vybereme otevřený vrchol v
8.   For sousedy s vrcholu v
9.   If $h(s) > h(v) + l(vs)$ # porovnáme délku nové a staré cesty
10. $h(s) \leftarrow h(v) + l(vs)$ # nastavíme kratší délku
11. $P(s) \leftarrow v$ # nastavíme předchůdce na nejkratší cestě
12.   pokud vrchol s není otevřený, otevřeme ho (tj. vložíme do fronty, listu či haldy)
13.   uzavřeme vrchol v (tj. odebereme v z fronty, listu či haldy)
14. Výstup algoritmu: délky nejkratších cest $h(v)$, předchozí vrchol na nejkratší cestě $P(v)$

Řádky 8. – 12. nazýváme RELAXACÍ VRCHOLU v:

8. For sousedy s vrcholu v
9.   If $h(s) > h(v) + l(vs)$ # porovnáme délku nové a staré cesty
10. $h(s) \leftarrow h(v) + l(vs)$ # nastavíme kratší délku
11. $P(s) \leftarrow v$ # nastavíme předchůdce na nejkratší cestě
12.   pokud vrchol s není otevřený, otevřeme ho (tj. vložíme do fronty, listu či haldy)

VYTIŠTĚNÍ CESTY: (cestu tiskneme od koncového vrcholu v k počátečnímu vrcholu v_0)

1. Print v
2. $v \leftarrow P(v)$
3. While v != None
4.   Print(v)
5. $v \leftarrow P(v)$