ΕΛΛΗΝΙΚΗ ΔΗΜΟΚΡΑΤΙΑ
Εθνικόν και Καποδιστριακόν
Πανεπιστήμιον Αθηνών

Αλγόριθμοι και Πολυπλοκότητα

Ενότητα 3

Αλγόριθμοι Γραφημάτων

Dijkstra

N. M. Μισυρλής

Τμήμα Πληροφορικής και Τηλεπικοινωνιών,

Εύρεση Ελάχιστου Μονοπατιού

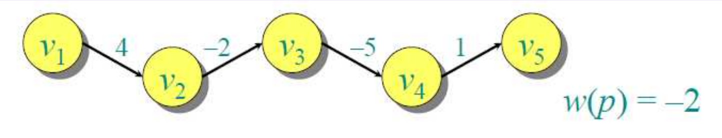
- Ιδιότητες ελάχιστου μονοπατιού
- Αλγόριθμος Dijkstra
- Πολυπλοκότητα
- Ορθότητα

Μονοπάτια σε γραφήματα

- Έστω γράφημα $G = (V, E)$ με συνάρτηση βάρους $w : E \to \mathbb{R}$. Τότε το βάρος ενός μονοπατιού $p = v_1 \to v_2 \to \ldots \to v_k$ ορίζεται ως εξής:

$$w(p) = \sum_{i=1}^{k-1} w(v_i, v_{i+1}).$$

- Παράδειγμα



Ελάχιστα Μονοπάτια

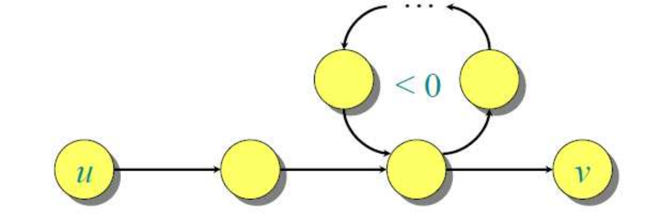
- Ελάχιστο μονοπάτι από τον κόμβο u στον v είναι ένα απλό μονοπάτι ελάχιστου βάρους από τον u στον v.
- Το βάρος του ελάχιστου μονοπατιού από τον u στον v ορίζεται ως

$$\delta(u,v) = min\{w(p) : p \text{ είναι ένα μονοπάτι από τον } u \text{ στον } v.\}$$

- $\delta(u,v) = \infty$ αν δεν υπάρχει μονοπάτι από τον u στον v.

Καλά Ορισμένα Ελάχιστα Μονοπάτια

- Αν ένα γράφημα G περιέχει κύκλο αρνητικού βάρους, τότε κάποια ελάχιστα μονοπάτια μπορεί να μην υπάρχουν.
- Παράδειγμα



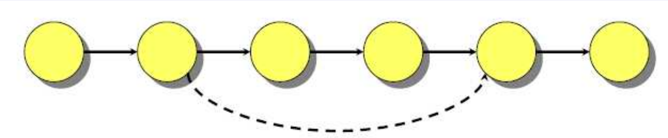
- $\delta(s, v) = -\infty$

Βέλτιστη Υποδομή

Theorem

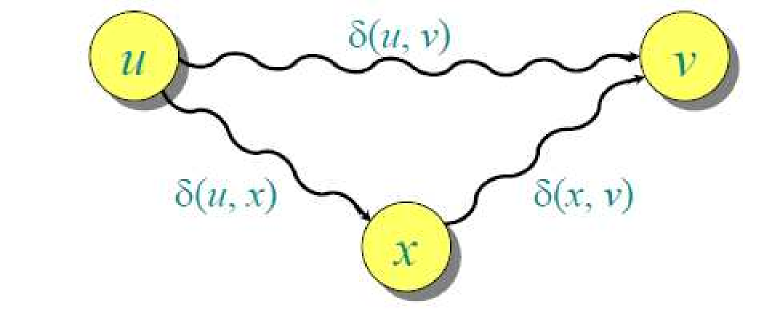
Ένα υπο-μονοπάτι ενός ελάχιστου μονοπατιού είναι ένα ελάχιστο μονοπάτι.

Κόψε και επικόλλησε: (cut and paste)



Τριγωνική Ανισότητα

- Έστω G ένα εμβαρές, κατευθυντό γράφημα με συνάρτηση βάρους w. Τότε
- $\delta(u, v) \leq \delta(u, x) + \delta(x, v)$



Ελάχιστα Μονοπάτια

- Πρόβλημα: Δεδομένου ενός αφετηριακού κόμβου $s \in V$, να βρεθούν τα βάρη των ελάχιστων μονοπατιών $\delta(s, v)$ για όλους τους $v \in V$
- Αν όλες οι ακμές έχουν μη αρνητικά βάρη $w(u, v)$, τότε όλα τα ελάχιστα μονοπάτια πρέπει να υπάρχουν.
- Ιδέα: Απληστία.
  1. Διατήρησε ένα συνόλο S κόμβων, των οποίων τα ελάχιστα μονοπάτια από τον s είναι γνωστά.
  2. Σε κάθε βήμα πρόσθεσε στο S τον κόμβο $v \in V - S$, του οποίου η εκτίμηση για την απόστασή του από τον s είναι ελάχιστη.
  3. Ενημέρωσε τις εκτιμήσεις των αποστάσεων των γειτονικών κόμβων του v.

Αλγόριθμος του Dijkstra

$d[s] \leftarrow 0$

for each $v \in V - \{s\}$

  do $d[v] \leftarrow \infty$

$S \leftarrow \emptyset$

$Q \leftarrow V$ /* Q είναι μια ουρά προτεραιότητας που διατηρεί τις $V - S$ */

while $Q \neq \emptyset$

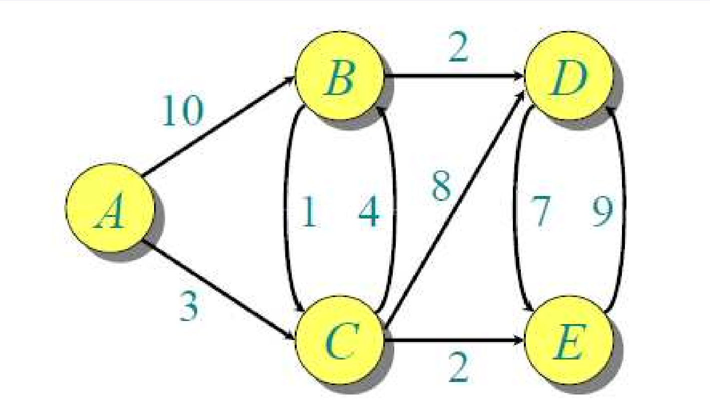
  do $u \leftarrow$ $\text{EXTRACT-MIN}(Q)$

    $S \leftarrow S \cup \{u\}$

    for each $v \in Adj[u]$

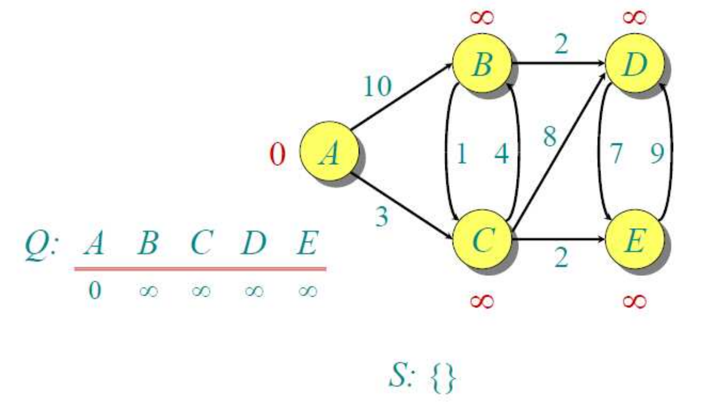
$$\left.\begin{array}{l} \text{do if } d[v] > d[u] + w(u,v) \\ \quad \text{then } d[v] \leftarrow d[u] + w(u,v) \end{array}\right\} \begin{array}{c} \textit{relaxation} \\ \textit{step} \end{array}$$

Παράδειγμα Αλγορίθμου Dijkstra

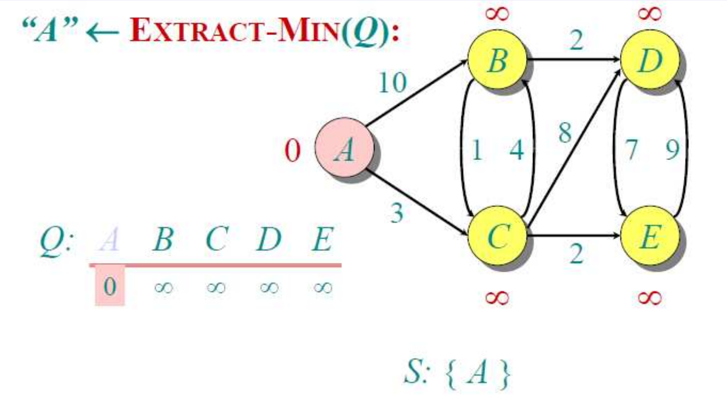


Παράδειγμα Αλγορίθμου Dijkstra

Αρχικοποίηση

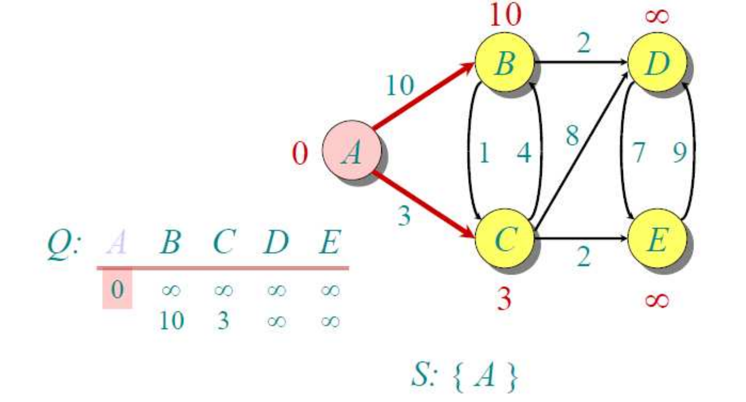


Παράδειγμα Αλγορίθμου Dijkstra



Παράδειγμα Αλγορίθμου Dijkstra

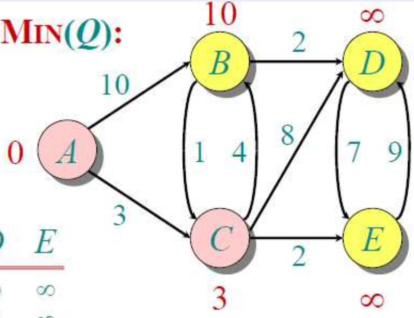
Χαλάρωση όλων των ακμών που ξεκινούν από την A.



Παράδειγμα Αλγορίθμου Dijkstra

“C” $\leftarrow$ **EXTRACT-MIN**(Q):

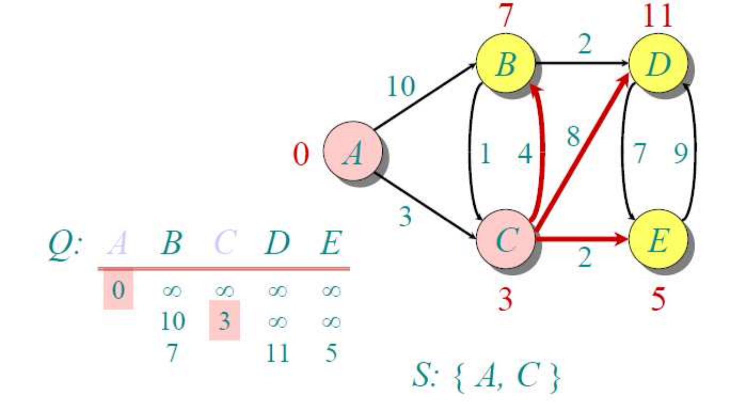
Q:	A	B	C	D	E
	0	∞	∞	∞	∞
		10	3	∞	∞



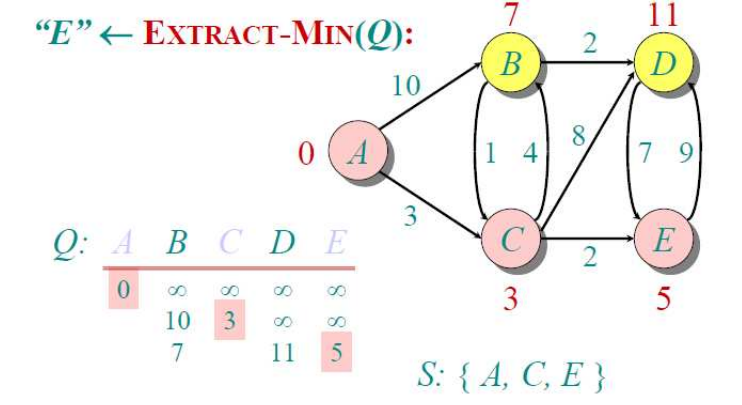
S: { A, C }

Παράδειγμα Αλγορίθμου Dijkstra

Χαλάρωση όλων των ακμών που ξεκινούν από την C.

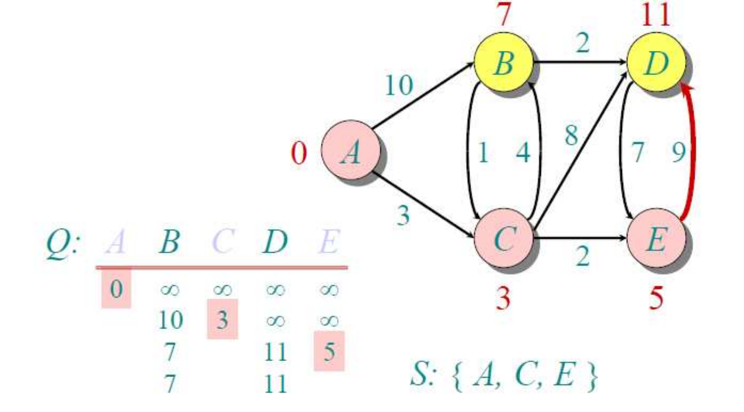


Παράδειγμα Αλγορίθμου Dijkstra

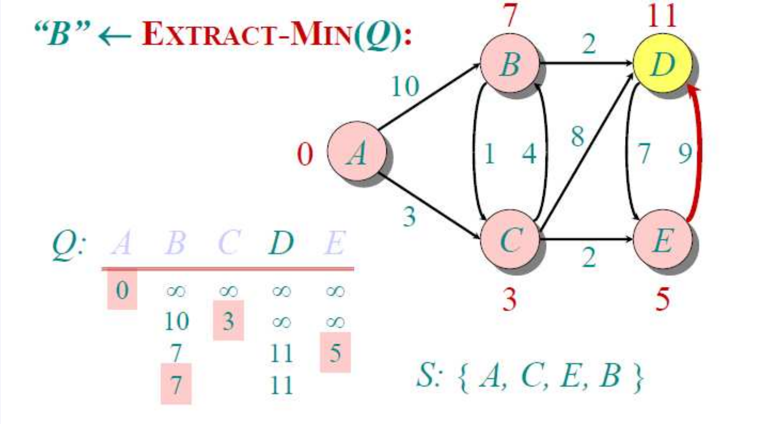


Παράδειγμα Αλγορίθμου Dijkstra

Χαλάρωση όλων των ακμών που ξεκινούν από την E.

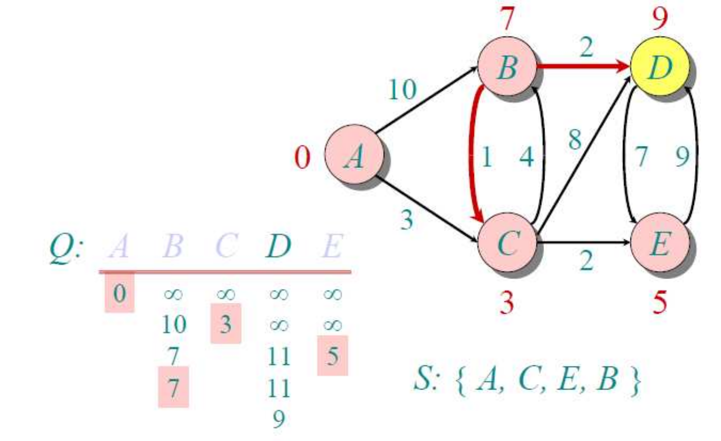


Παράδειγμα Αλγορίθμου Dijkstra



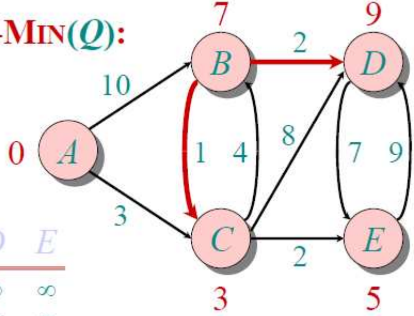
Παράδειγμα Αλγορίθμου Dijkstra

Χαλάρωση όλων των ακμών που ξεκινούν από την B.



Παράδειγμα Αλγορίθμου Dijkstra

“D” ← EXTRACT-MIN(Q):



Q:	A	B	C	D	E
	$\boxed{0}$	∞	∞	∞	∞
		10	$\boxed{3}$	∞	∞
		7		11	$\boxed{5}$
		$\boxed{7}$		11	
				$\boxed{9}$	

S: $\{ A, C, E, B, D \}$

Ορθότητα I

Lemma

Αρχικοποιώντας $d[s] \leftarrow 0$ *και* $d[v] \leftarrow \infty$ *για όλους τους* $v \in V$ *ισχύει ότι* $d[v] \geq \delta(s, v)$ *για όλους τους* $v \in V$. *Η συνθήκη αυτή διατηρείται για οποιαδήποτε ακολουθία πράξεων χαλάρωσης στις ακμές του* G. *Επιπλέον, από την στιγμή που* $d[v] = \delta(s, v)$ *η* $d[v]$ *δεν μεταβάλλεται ποτέ.*

Proof.

Με επαγωγή ως προς το πλήθος των πράξεων χαλάρωσης. Βάση.

$$d[s] = 0 \geq \delta(s, s) \quad \delta(s, v) \leq \infty = d[v]$$

Επαγωγικό βήμα. Χαλάρωση της ακμής (u, v). Από την επαγωγική υπόθεση, έπεται ότι πριν από την πράξη χαλάρωσης ισχύει

$$d[x] \geq \delta(s, x) \quad x \in V$$

□

Ορθότητα I

$$
\begin{aligned}
d[v] &= d[u] + w(u, v) && \text{χαλάρωση} \\
&\geq \delta(s, u) + w(u, v) && \text{επαγωγική υπόθεση} \\
&= \delta(s, u) + \delta(u, v) \\
&\geq \delta(s, v) && \text{τριγωνική ανισότητα} .
\end{aligned}
$$

Ορθότητα II

Lemma

Έστω u *ο προκάτοχος του* v *στο ελάχιστο μονοπάτι από τον* s *στον* v. *Τότε, αν* $d[u] = \delta(s, u)$ *και εφαρμοστεί η χαλάρωση στην ακμή* (u, v) *, έχουμε* $d[v] = \delta(s, v)$ *μετά την χαλάρωση.*

Proof.

Από το προηγούμενο Λήμμα έχουμε ότι $d[v] \geq \delta(s, v)$ πριν την χαλάρωση. Αν $d[v] = \delta(s, v)$ τελειώσαμε.
Αν $d[v] > \delta(s, v)$. Τότε, αν $d[v] > d[u] + w(u, v)$ εφαρμόζεται χαλάρωση και

$$d[v] = d[u] + w(u, v)$$

$$= \delta(s, u) + w(u, v)$$

λόγω της υπόθεσης

$$= \delta(s, v)$$

□

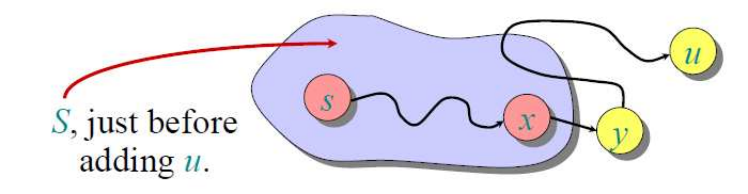
Ορθότητα III

Theorem

Ο αλγόριθμος του Dijkstra τερματίζει με $d[v] = \delta(s, v)$ *για όλους τους* $v \in V$.

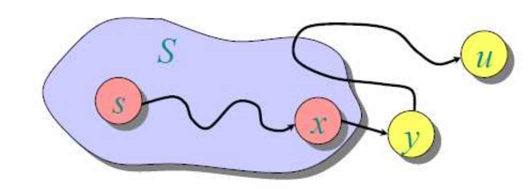
Proof.

Αρκεί να δείξουμε ότι $d[v] = \delta(s, v)$ για κάθε $v \in V$ όταν ο v προστίθεται στο S. Έστω u ο πρώτος κόμβος που εισάγεται στο S και $d[u] > \delta(s, u)$. Έστω y ο πρώτος κόμβος στο $V - S$ κατά μήκος του ελάχιστου μονοπατιού από τον s στον u και x ο προκάτοχός του:



□

Ορθότητα III (συν.)



Αφού ο u είναι ο πρώτος κόμβος που παραβιάζει τον ισχυρισμό μας, έχουμε $d[x] = \delta(s, x)$. Όταν ο x προστέθηκε στο S, εφαρμόστηκε η χαλάρωση στην ακμή (x, y) , το οποίο συνεπάγεται

$$d[y] = \delta(s, y).$$

Ο κόμβος y προηγείται του u και όλα τα βάρη των ακμών είναι μη αρνητικά, συνεπώς

$$\delta(s, y) \leq \delta(s, u).$$

Ορθότητα III (συν.)

Συνοψίζοντας έχουμε τελικά ότι

$$d[y] = \delta(s, y) \leq \delta(s, u) \leq d[u]$$

όπου η τελευταία προκύπτει από την ιδιότητα του άνω φράγματος.

Αλλά, η επιλογή του u έγινε διότι

$$d[u] \leq d[y]$$

Συνεπώς οι ανωτέρω ανισότητες είναι ισότητες, δηλαδή

$$d[y] = \delta(s, y) = \delta(s, u) = d[u]$$

Τελικά

$$d[u] = \delta(s, u)$$

που αντιβαίνει στην υπόθεσή μας.

Ανάλυση του Dijkstra

$$
\begin{array}{ll}
\begin{array}{c}|V| \\ \textit{times}\end{array} &
\left\{
\begin{array}{l}
\quad\text{while } Q \neq \emptyset \\
\qquad\text{do } u \leftarrow \textsc{Extract-Min}(Q) \\
\qquad\quad S \leftarrow S \cup \{u\} \\
\begin{array}{c}\textit{degree}(u) \\ \textit{times}\end{array}
\left\{
\begin{array}{l}
\text{for each } v \in \textit{Adj}[u] \\
\quad\text{do if } d[v] > d[u] + w(u,v) \\
\qquad\text{then } d[v] \leftarrow d[u] + w(u,v)
\end{array}
\right.
\end{array}
\right.
\end{array}
$$

- $\Theta(E)$ ακριβώς DECREASE-KEY's.
- Χρόνος = $\Theta(V \cdot T_{\textsc{Extract-Min}} + E \cdot T_{\textsc{Decrease-Key}})$
- Ίδια όπως στην ανάλυση του αλγορίθμου του Prim.

Ανάλυση του Dijkstra (συν.)

$$\text{Χρόνος} = \Theta(V \cdot T_{\textsc{Extract-Min}} + E \cdot T_{\textsc{Decrease-Key}})$$

Q	$T_{\textsc{Extract-Min}}$	$T_{\textsc{Decrease-Key}}$	Συνολικό
array	$O(V)$	$O(1)$	$O(V^2)$
binary Heap	$O(\log V)$	$O(\log V)$	$O(E \log V)$
fibonacci Heap	$O(\log V)$	$O(1)$	$O(E + V \log V)$

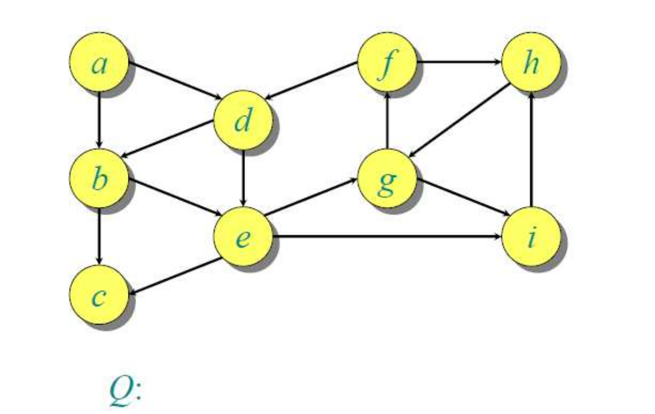
Γράφοι χωρίς Βάρη

- Θεωρήστε $w(u, v) = 1$ για όλες τις $(u, v) \in E$.
- Μπορεί να βελτιωθεί ο αλγόριθμος του Dijkstra ;
  - Χρησιμοποίησε μια ουρά FIFO ant'i gia our'a proterai'othtac
- Αναζήτηση Κατά-Πλάτος

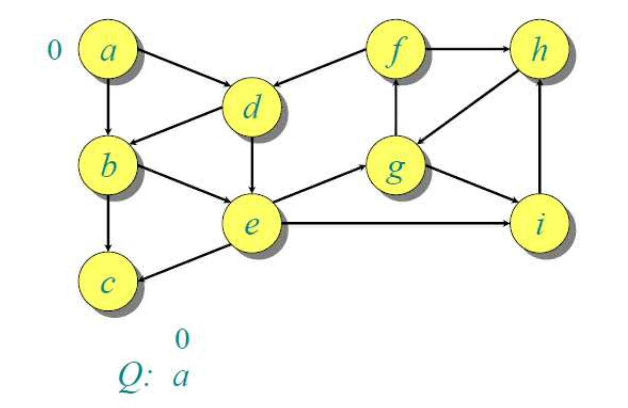
$$
\begin{array}{l}
\text{while } Q \neq \emptyset \\
\quad \text{do } u \leftarrow \textsc{Dequeue}(Q) \\
\quad\quad \text{for each } v \in Adj[u] \\
\quad\quad\quad \text{do if } d[n] = \infty \\
\quad\quad\quad\quad \text{then } d[v] \leftarrow d[u] + 1 \\
\quad\quad\quad\quad\quad \textsc{Enqueue}(Q, v)
\end{array}
$$

- Χρόνος = $O(V + E)$.

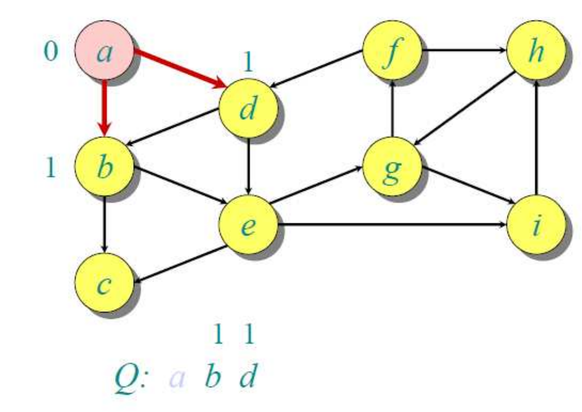
Παράδειγμα Αλγορίθμου Αναζήτησης Κατά-Πλάτος



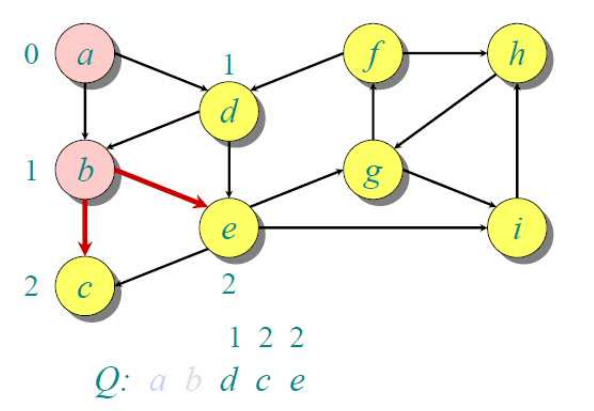
Παράδειγμα Αλγορίθμου Αναζήτησης Κατά-Πλάτος



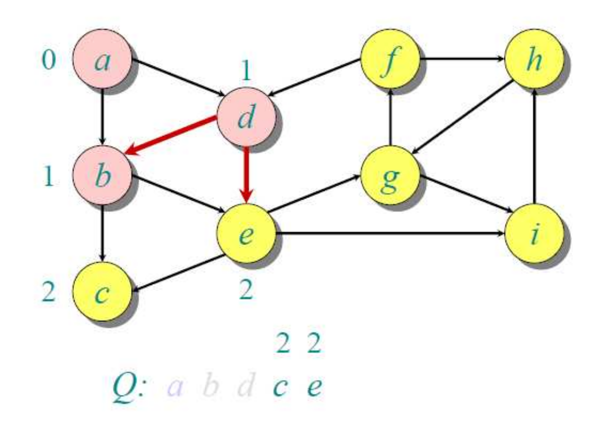
Παράδειγμα Αλγορίθμου Αναζήτησης Κατά-Πλάτος



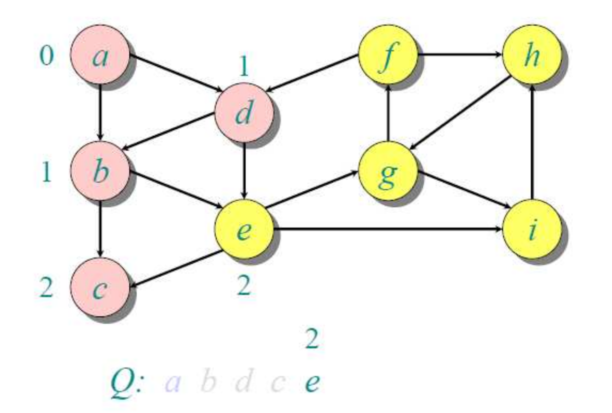
Παράδειγμα Αλγορίθμου Αναζήτησης Κατά-Πλάτος



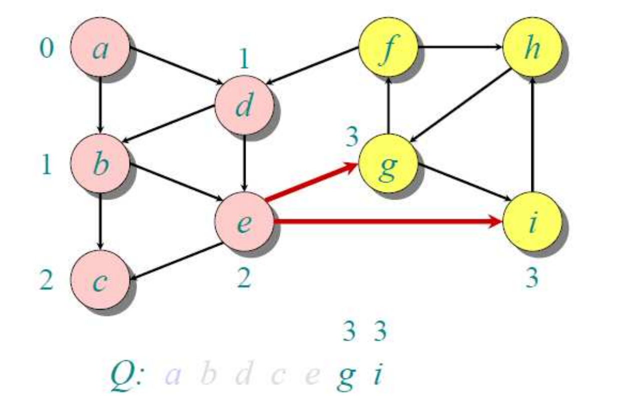
Παράδειγμα Αλγορίθμου Αναζήτησης Κατά-Πλάτος



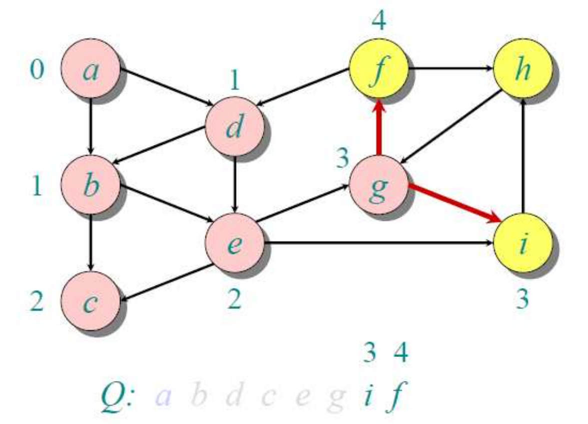
Παράδειγμα Αλγορίθμου Αναζήτησης Κατά-Πλάτος



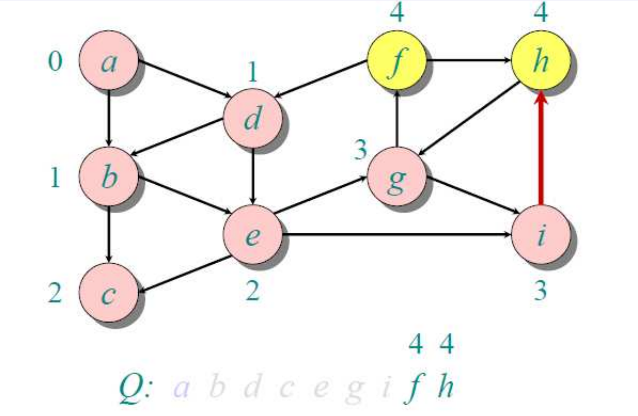
Παράδειγμα Αλγορίθμου Αναζήτησης Κατά-Πλάτος



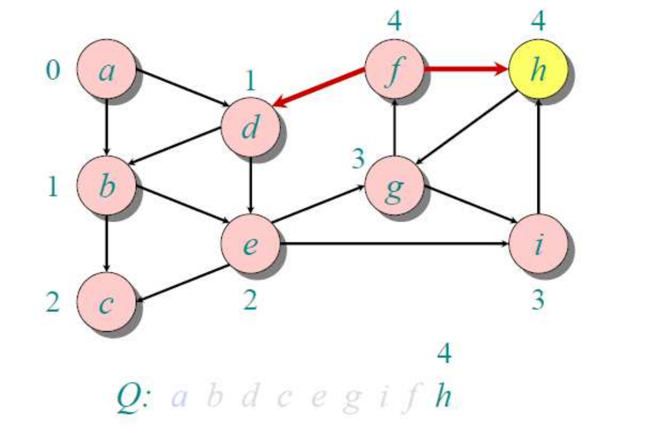
Παράδειγμα Αλγορίθμου Αναζήτησης Κατά-Πλάτος



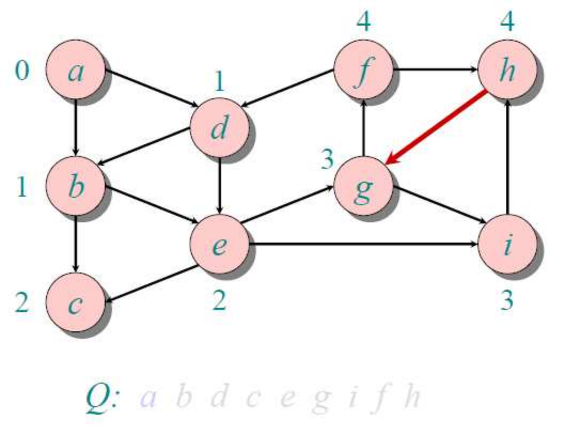
Παράδειγμα Αλγορίθμου Αναζήτησης Κατά-Πλάτος



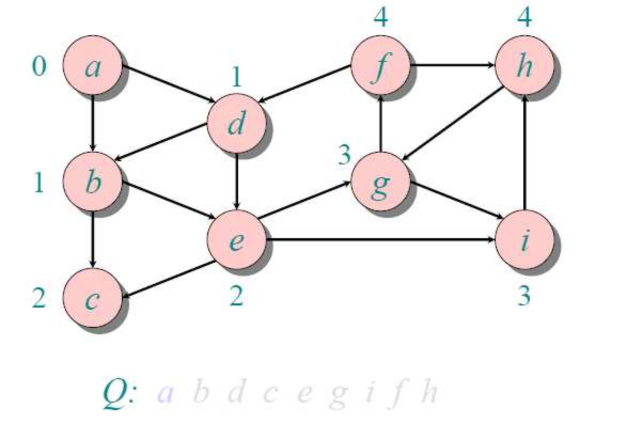
Παράδειγμα Αλγορίθμου Αναζήτησης Κατά-Πλάτος



Παράδειγμα Αλγορίθμου Αναζήτησης Κατά-Πλάτος



Παράδειγμα Αλγορίθμου Αναζήτησης Κατά-Πλάτος



Ορθότητα του Αλγορίθμου Αναζήτησης Κατά-Πλάτος

while $Q \neq \emptyset$
  do $u \leftarrow \text{DEQUEUE}(Q)$
    for each $v \in Adj[u]$
      do if $d[n] = \infty$
        then $d[v] \leftarrow d[u] + 1$
          $\text{ENQUEUE}(Q, v)$

- **Βασική Ιδέα:** Η ουρά FIFO Q στην αναζήτηση κατά πλάτος μιμείται την ουρά προτεραιότητας Q του Dijkstra
- Το γεγονός ότι η v έρχεται μετά την u στην Q συνεπάγεται ότι $d[v] = d[u]$ ή $d[v] = d[u] + 1$.

Σημειώματα

Σημείωμα Αναφοράς

Copyright Εθνικόν και Καποδιστριακόν Πανεπιστήμιον Αθηνών 2015, Νικόλαος Μισυρλής, "Αλγόριθμοι και Πολύπλοκότητα. Ενότητα 3 - Αλγόριθμοι Γραφημάτων" Έκδοση:1.01 . Αθήνα 2015. Διαθέσιμο από τη δικτυακή διεύθυνση:`http://opencourses.uoa.gr/courses/DI13/` .

Σημείωμα Αδειοδότησης

Διατήρηση Σημειωμάτων

Οποιαδήποτε αναπαραγωγή ή διασκευή του υλικού θα πρέπει να συμπεριλαμβάνει:

- το Σημείωμα Αναφοράς
- το Σημείωμα Αδειοδότησης
- τη δήλωση Διατήρησης Σημειωμάτων
- το Σημείωμα Χρήσης Έργων Τρίτων (εφόσον υπάρχει)

μαζί με τους συνοδευόμενους υπερσυνδέσμους.