ΑΡΙΣΤΟΤΕΛΕΙΟ
ΠΑΝΕΠΙΣΤΗΜΙΟ
ΘΕΣΣΑΛΟΝΙΚΗΣ

ΑΝΟΙΚΤΑ
ΑΚΑΔΗΜΑΪΚΑ
ΜΑΘΗΜΑΤΑ

ΑΛΓΟΡΙΘΜΟΙ

Ενότητα 5: Μείωσε και Βασίλευε

Ιωάννης Μανωλόπουλος, Καθηγητής
Αναστάσιος Γούναρης, Επίκουρος Καθηγητής
Τμήμα Πληροφορικής ΑΠΘ

Ευρωπαϊκή Ένωση
Ευρωπαϊκό Κοινωνικό Ταμείο
ΕΠΙΧΕΙΡΗΣΙΑΚΟ ΠΡΟΓΡΑΜΜΑ
ΕΚΠΑΙΔΕΥΣΗ ΚΑΙ ΔΙΑ ΒΙΟΥ ΜΑΘΗΣΗ
επένδυση στην κοινωνία της γνώσης
ΥΠΟΥΡΓΕΙΟ ΠΑΙΔΕΙΑΣ ΚΑΙ ΘΡΗΣΚΕΥΜΑΤΩΝ
ΕΙΔΙΚΗ ΥΠΗΡΕΣΙΑ ΔΙΑΧΕΙΡΙΣΗΣ
ΕΣΠΑ 2007-2013
πρόγραμμα για την ανάπτυξη
ΕΥΡΩΠΑΪΚΟ ΚΟΙΝΩΝΙΚΟ ΤΑΜΕΙΟ
Με τη συγχρηματοδότηση της Ελλάδας και της Ευρωπαϊκής Ένωσης

Άδειες Χρήσης

- Το παρόν εκπαιδευτικό υλικό υπόκειται σε άδειες χρήσης Creative Commons.
- Για εκπαιδευτικό υλικό, όπως εικόνες, που υπόκειται σε άλλου τύπου άδειας χρήσης, η άδεια χρήσης αναφέρεται ρητώς.

Χρηματοδότηση

- Το παρόν εκπαιδευτικό υλικό έχει αναπτυχθεί στα πλαίσια του εκπαιδευτικού έργου του διδάσκοντα.
- Το έργο «Ανοικτά Ακαδημαϊκά Μαθήματα στο Αριστοτέλειο Πανεπιστήμιο Θεσσαλονίκης» έχει χρηματοδοτήσει μόνο τη αναδιαμόρφωση του εκπαιδευτικού υλικού.
- Το έργο υλοποιείται στο πλαίσιο του Επιχειρησιακού Προγράμματος «Εκπαίδευση και Δια Βίου Μάθηση» και συγχρηματοδοτείται από την Ευρωπαϊκή Ένωση (Ευρωπαϊκό Κοινωνικό Ταμείο) και από εθνικούς πόρους.

Ευρωπαϊκή Ένωση
Ευρωπαϊκό Κοινωνικό Ταμείο
ΕΠΙΧΕΙΡΗΣΙΑΚΟ ΠΡΟΓΡΑΜΜΑ
ΕΚΠΑΙΔΕΥΣΗ ΚΑΙ ΔΙΑ ΒΙΟΥ ΜΑΘΗΣΗ
επένδυση στην κοινωνία της γνώσης
ΥΠΟΥΡΓΕΙΟ ΠΑΙΔΕΙΑΣ & ΘΡΗΣΚΕΥΜΑΤΩΝ, ΠΟΛΙΤΙΣΜΟΥ & ΑΘΛΗΤΙΣΜΟΥ
ΕΙΔΙΚΗ ΥΠΗΡΕΣΙΑ ΔΙΑΧΕΙΡΙΣΗΣ
ΕΣΠΑ 2007-2013
πρόγραμμα για την ανάπτυξη
ΕΥΡΩΠΑΪΚΟ ΚΟΙΝΩΝΙΚΟ ΤΑΜΕΙΟ
Με τη συγχρηματοδότηση της Ελλάδας και της Ευρωπαϊκής Ένωσης

ΑΡΙΣΤΟΤΕΛΕΙΟ
ΠΑΝΕΠΙΣΤΗΜΙΟ
ΘΕΣΣΑΛΟΝΙΚΗΣ

ΑΝΟΙΚΤΑ
ΑΚΑΔΗΜΑΪΚΑ
ΜΑΘΗΜΑΤΑ

Μείωσε και Βασίλευε

Μείωση κατά ένα, Μείωση κατά ένα σταθερό όρο, Μείωση κατά μεταβλητό μέγεθος

Μείωσε και Βασίλευε

1. Μειώνουμε το στιγμιότυπο του προβλήματος σε ένα μικρότερο στιγμιότυπο του ιδίου προβλήματος
2. Επιλύουμε το μικρότερο στιγμιότυπο
3. Επεκτείνουμε τη λύση του μικρότερου προβλήματος για να λύσουμε το αρχικό πρόβλημα

- Συναντάται και με τις δύο μορφές: «από κάτω προς τα πάνω (bottom-up» και «από πάνω προς τα κάτω (top-down)»
- Αναφέρεται επίσης και ως *επαγωγική (inductive)* ή *επαυξητική (incremental)* προσέγγιση

Παραδείγματα «Μείωσε και Βασίλευε»

- *Μείωση κατά ένα:*
  - Ταξινόμηση με εισαγωγή
  - Αλγόριθμοι αναζήτησης σε γράφους:
    - DFS, BFS, τοπολογική ταξινόμηση
  - Αλγόριθμοι για τη δημιουργία διατάξεων, υποσυνόλων
- *Μείωση κατά ένα σταθερό όρο:*
  - Δυαδική αναζήτηση
  - Προβλήματα κίβδηλων νομισμάτων
  - Πολλαπλασιασμός αλά ρωσικά
  - Πρόβλημα του Josephus
- *Μείωση κατά μεταβλητό μέγεθος:*
  - Αλγόριθμος του Ευκλείδη
  - Επιλογή με διαμέριση

Ποιά είναι η διαφορά;

Έστω το πρόβλημα της ύψωσης σε δύναμη: a^n

- Ωμή Βία: $a*a*a*a*...*a$
- Διαίρει και Βασίλευε: $a^{n/2} * a^{n/2}$
- Μείωσε κατά ένα: $a^{n-1} * a$
- Μείωσε κατά ένα σταθερό όρο: $(a^{n/2})^2$

Ταξινόμηση με εισαγωγή

Algorithm InsertionSort(A[0..n-1])

```
// Sorts a given array by insertion sort
// Input: An array A[0..n-1] of n orderable elements
// Output: Array A[0..n-1] sorted in nondecreasing
order
for i ← 1 to n-1 do
      v ← A[i]; j ← i-1;
      while j ≥ 0 and A[j] > v do
            A[j+1] ← A[j]; j ← j-1;
      A[j+1] ← v
```

- Βελτιώσεις: κόμβος φρουρός, δυαδική εισαγωγή, ταξινόμηση του Shell
- Αποδοτικότητα ?

Ανάλυση της ταξινόμησης με εισαγωγή

- Χρονική αποδοτικότητα

  $C_{worst}(n) = n(n\text{-}1)/2 \in \Theta(n^2)$

  $\mathrm{C}_{\mathrm{avg}}(n) \approx n^2/4 \in \Theta(n^2)$

  $C_{best}(n) = n - 1 \in \Theta(n)$ (είναι γρήγορη και για <u>σχεδόν</u> ταξινομημένους πίνακες)

- Χωρική αποδοτικότητα: επιτόπιος αλγόριθμος (in-place)

- Ευστάθεια: ναι

Ταξινόμηση με εισαγωγή - παράδειγμα

Παράδειγμα: Ταξινόμησε τους αριθμούς 6, 4, 1, 8, 5

6 | $\underline{4}$ 1 8 5

4 6 | $\underline{1}$ 8 5

1 4 6 | $\underline{8}$ 5

1 4 6 8 | $\underline{5}$

1 4 5 6 8

Διάσχιση γράφων

- Πολλά προβλήματα απαιτούν τη συστηματική επεξεργασία των κορυφών ενός γράφου

- *Αλγόριθμοι διάσχισης γράφου:*
  - Αναζήτηση κατά βάθος (depth-first search - DFS)
  - Αναζήτηση κατά πλάτος (breadth-first search - BFS)

Αναζήτηση κατά βάθος (DFS)

- Εξερευνούμε το γράφο $G=(V,E)$ κινούμενοι πάντοτε όσο το δυνατό μακρύτερα από την τελευταία επισκεφθείσα κορυφή
- Ομοιότητα με την προδιατεταγμένη διάσχιση δένδρου
- **Algorithm DFS(G)**

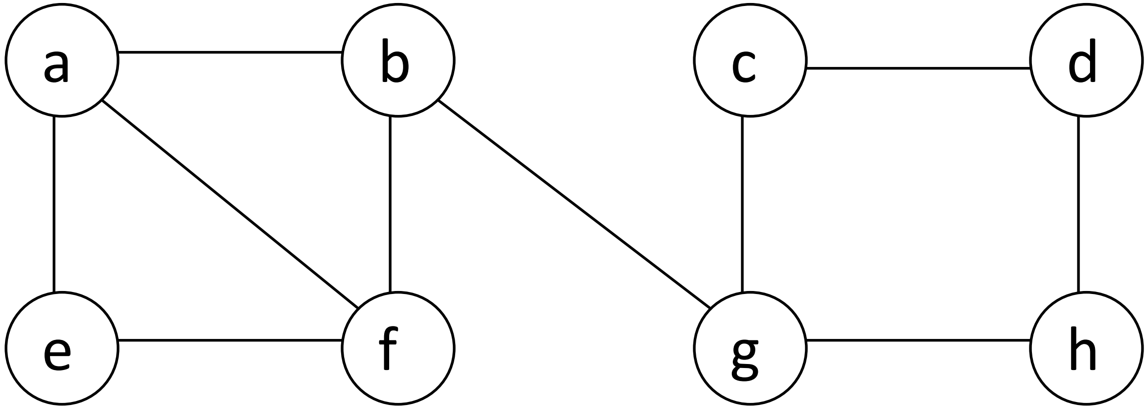
```
count :=0
mark each vertex with 0 (unvisited)
for each vertex v∈V do
   if v is marked with 0: dfs(v)
dfs(v)
count ← count + 1
mark v with count
for each vertex w adjacent to v do
   if w is marked with 0: dfs(w)
```

Αναζήτηση κατά βάθος (συνέχεια)

- Χρησιμοποιείται μία στοίβα
  - Ένας κόμβος ωθείται στην στοίβα όταν επισκέπτεται για πρώτη φορά.
  - Ένας κόμβος απωθείται από τη στοίβα όταν δεν υπάρχει γειτονικός κόμβος που δεν έχει επισκεπτεί, δηλ. απωθείται ένας κόμβος όταν καθίσταται αδιέξοδο.

Παράδειγμα DFS – Μη κατευθυνόμενος γράφος

- Σειρά επίσκεψης με αναζήτηση κατά βάθος;



Τύποι ακμών σε ένα δάσος DFS

- *Δενδρικές ακμές (tree edges):* ακμές που σχηματίζουν το δένδρο
- *Οπίσθιες ακμές (back edges):* ακμές προς κόμβους-προγόνους

Αναζήτηση κατά βάθος

- Η DFS μπορεί να υλοποιηθεί με γράφους που αναπαρίστανται:
  - Πίνακα γειτνίασης: Θ(V^2)
  - Συνδεδεμένες λίστες γειτνίασης: Θ(V+E)
- Δίνει δύο διακριτές διατάξεις κορυφών:
  - καθώς οι κορυφές συναντώνται για πρώτη φορά (όταν ωθούνται σε μία στοίβα)
  - καθώς οι κορυφές συναντώνται για δεύτερη φορά (όταν απωθούνται από τη στοίβα)
- Εφαρμογές (Hopcroft-Tarjan):
  - Έλεγχος συνδεσιμότητας, εύρεση συνδεδεμένων συνιστωσών
  - Έλεγχος κύκλων, σημεία συναρμογής
  - Εύρεση του χώρου καταστάσεων του προβλήματος προς επίλυση (AI)

Αναζήτηση κατά πλάτος: σημειώσεις

- Εξερευνούμε το γράφο επισκεπτόμενοι όλους του γείτονες του τελευταίου επισκεφθέντος κόμβου
- Παρόμοια με τη διάσχιση κατά επίπεδα
- Χρήση ουράς (αντί στοίβα)
- Εφαρμογές: όπως DFS, αλλά επίσης μπορούμε να βρούμε μονοπάτια από κόμβο σε κόμβο με το μικρότερο αριθμό ακμών

Αλγόριθμος αναζήτησης κατά πλάτος (1)

Algorithm BFS(G)

```
// Implements a breadth first search traversal
of a given graph

// Input: Graph G = (V,E)

// Output: Graph G with its vertices marked
with consecutive integers in the order they've
been first encountered by the BFS traversal

mark each vertex in V with 0 to denoted
"unvisited" count <- 0

for each vertex v in V do

     if v is marked with 0

          bfs(v)
```

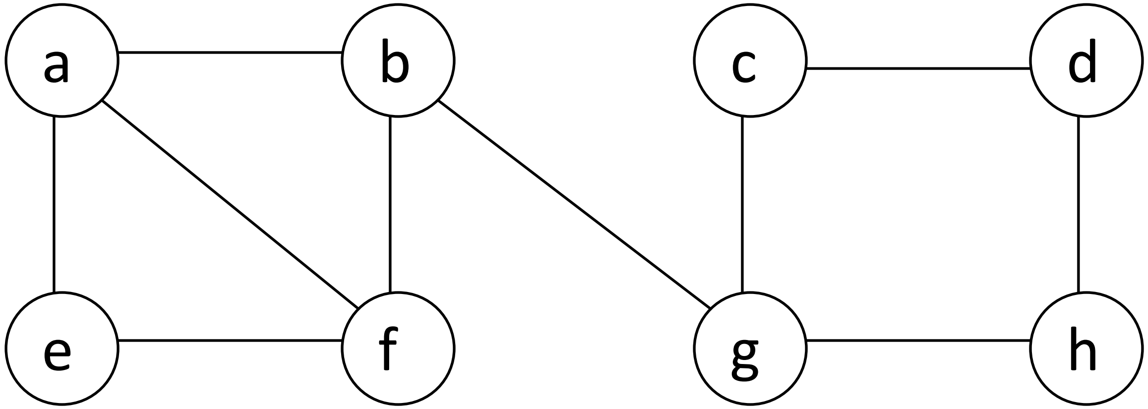
Αλγόριθμος αναζήτησης κατά πλάτος (2)

bfs(v)

```
// visits all the unvisited vertices connected to vertex v
by a path and assigns them the numbers in the order they
are visited via global variable count

count <- count + 1;

mark v with count and initialize a queue with v

while the queue is not empty do

    for each vertex w in V adjacent to the front vertex do

            if w is marked with 0

                  count <- count + 1; mark w with count

                  add w to the queue

    remove the front vertex from the queue
```

Παράδειγμα BFS – Μη κατευθυνόμενος γράφος

- Σειρά επίσκεψης με αναζήτηση κατά πλάτος;



Αναζήτηση κατά πλάτος: σημειώσεις

- Η BFS έχει την ίδια αποτελεσματικότητα με τη DFS αναλόγως αν ο γράφος υλοποιείται με:
  - Πίνακες γειτνίασης: Θ(V^2)
  - Συνδεδεμένες λίστες γειτνίασης: Θ(V+E)
- Δίνει μόνο μία διάταξη των κορυφών (η σειρά εισαγωγής/εξαγωγής από την ουρά είναι ίδια)

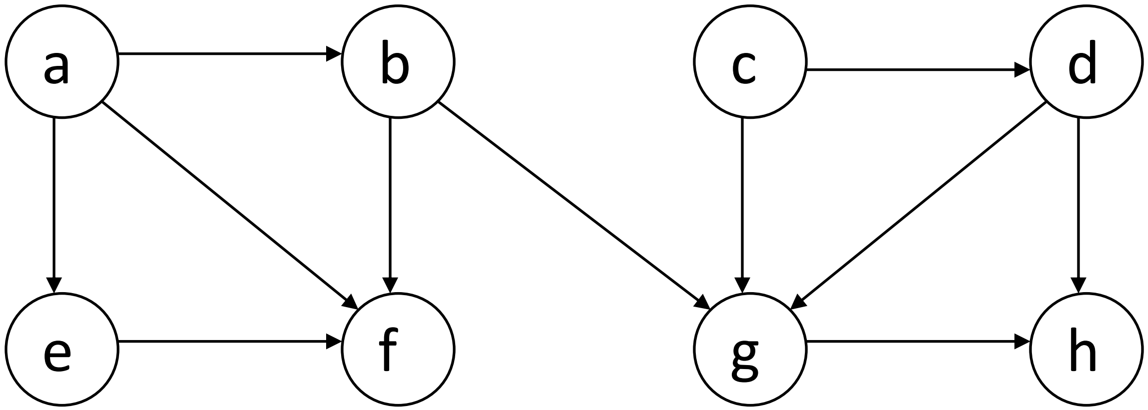
Κατευθυνόμενος άκυκλος γράφος (dag)

- Dag είναι ένας κατευθυνόμενος γράφος χωρίς κύκλους
- Προκύπτουν σε πολλά προβλήματα όπως:
  - Δομές με προαπαιτούμενα
  - Τροφικές αλυσίδες
- Βασίζονται σε μερικές διατάξεις του πεδίου
- Τοπολογική ταξινόμηση:

  να βρεθεί μία διάταξη κορυφών για την οποία κάθε ακμή να εξέρχεται από μία κορυφή που εμφανίζεται νωρίτερα από την κορυφή που εισέρχεται.

Παράδειγμα DFS– κατευθυνόμενος γράφος

- Σειρά επίσκεψης με αναζήτηση κατά βάθος;

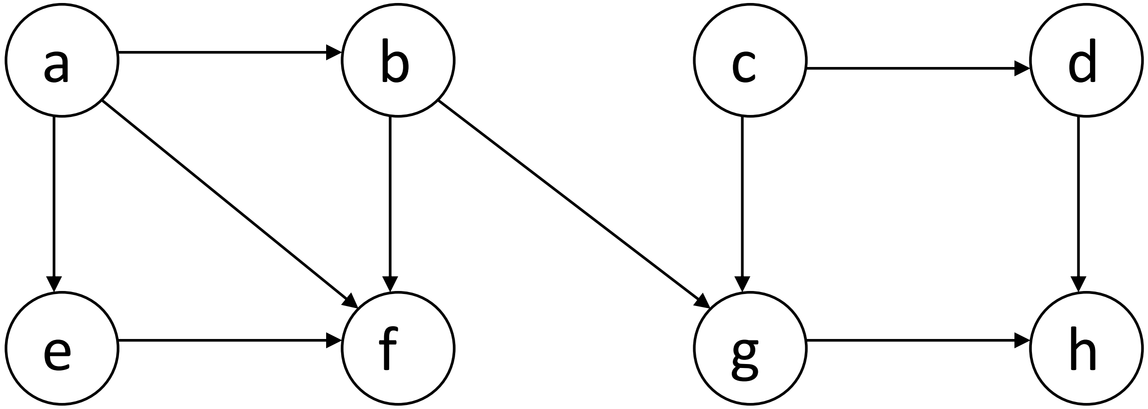


Τύποι ακμών σε ένα δάσος DFS

- *Δενδρικές ακμές (tree edges):* ακμές που σχηματίζουν το δένδρο
- *Οπίσθιες ακμές (back edges):* ακμές προς κόμβους-προγόνους
- *Εμπρόσθιες ακμές (forward edges)*: ακμές προς κόμβους-απογόνους
- *Διασταυρούμενες ακμές (cross edges):* καμία περίπτωση από τις ανωτέρω

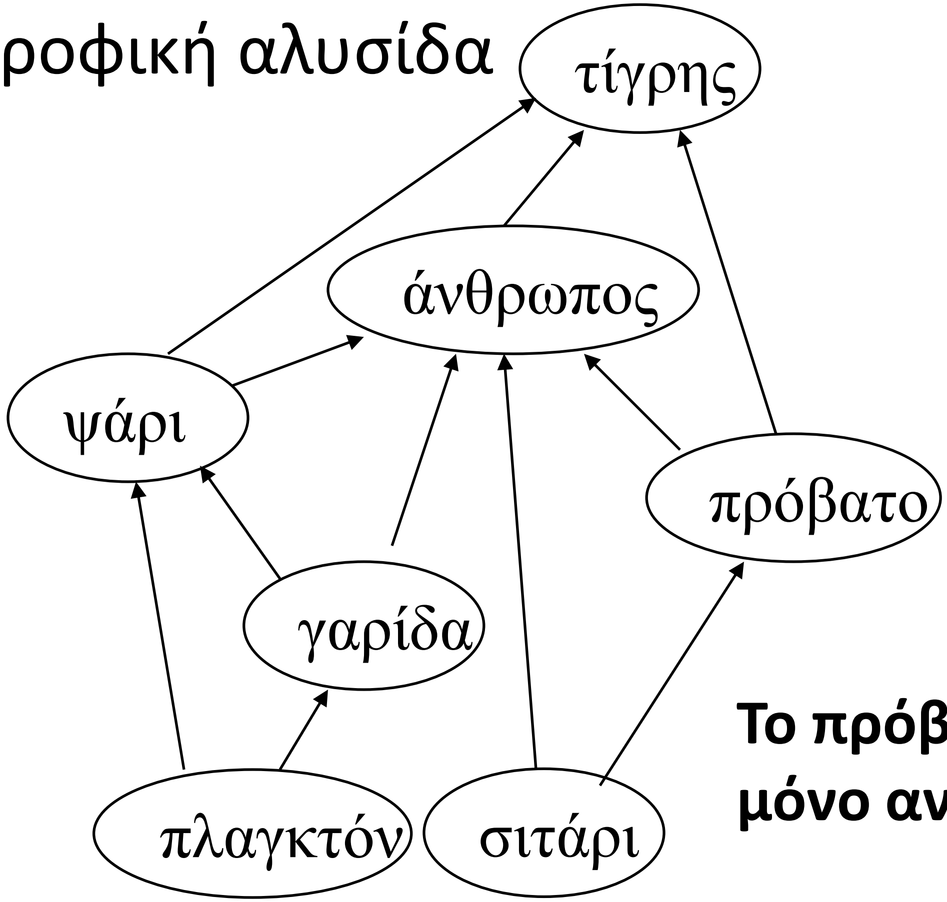
Παράδειγμα – Κατευθυνόμενος γράφος

- Σειρά επίσκεψης με αναζήτηση κατά πλάτος;



Τοπολογική ταξινόμηση

- Να βρεθεί μία συνολική διάταξη συμβατή με την τροφική αλυσίδα



Το πρόβλημα επιλύεται αν και μόνο αν ο γράφος είναι dag

Αλγόριθμοι τοπολογικής ταξινόμησης

1. Αλγόριθμοι βασισμένοι σε DFS:
   - Διασχίζουμε με DFS και καταγράφουμε τη διάταξη με την οποία οι κόμβοι απωθούνται από τη στοίβα
   - Αναστρέφοντας τη διάταξη επιλύεται το πρόβλημα
   - Αν συναντηθούν οπίσθιες ακμές → δεν είναι dag!
2. Αλγόριθμος με διαγραφή πηγής (source)
   - Διαδοχικά βρίσκει και απομάκρυνει μία πηγή (ένας κόμβος χωρίς εισερχόμενες ακμές) μαζί με όλες τις εξερχόμενες ακμές της

- Κοινή επίδοση Θ(*V*+E) με συνδεδεμένες λίστες γειτνίασης

Δημιουργία μεταθέσεων

- Χρήσιμη σε εξαντλητικούς αλγορίθμους αναζήτησης
- Απαίτηση ελάχιστης μεταβολής (minimal-change requirement)
- Δίνεται: ένα σύνολο μεταθέσεων *k*-1 αντικειμένων
- Μέθοδος: Δημιουργία μίας νέας μετάθεσης εισάγοντας το *k*-οστό αντικείμενο σε όλες τις δυνατές θέσεις μετακινούμενοι κατά μία κατεύθυνση (από τα δεξιά προς τα αριστερά) για την $1^η$ από τις μεταθέσεις μεγέθους *k*-1. Στη συνέχεια, για την επόμενη μετάθεση μεγέθους *k*-1, εισάγουμε το *k*-οστό αντικείμενο εναλλάσσοντας την κατεύθυνση (από τα αριστερά προς τα δεξιά)

Δημιουργία μεταθέσεων – Johnson-Trotter

Τέχνασμα: ένας βέλος δείχνει την κατεύθυνση ενός αντικειμένου. Ένα αντικείμενο ονομάζεται *κινητό* (*mobile)* αν δείχνει σε ένα μικρότερο αντικείμενο. Δεν δημιουργούνται στιγμιότυπα μικρότερου μεγέθους.

Αλγόριθμος Johnson-Trotter

Initialize the first permutation $\overleftarrow{1}\ \overleftarrow{2}\ \overleftarrow{3}\ \ldots\ \overleftarrow{n}$

while there exist a mobile integer *k* do

- **find the largest mobile integer *k***
- **swap *k* and the adjacent integer its arrow points to**
- **reverse the direction of all integers that are larger than *k***

Παράδειγμα $\overleftarrow{1}\overleftarrow{2}\overleftarrow{3}$, $\overleftarrow{1}\overleftarrow{3}\overleftarrow{2}$, $\overleftarrow{3}\overleftarrow{1}\overleftarrow{2}$, $\overrightarrow{3}\overleftarrow{2}\overleftarrow{1}$, $\overleftarrow{2}\overrightarrow{3}\overleftarrow{1}$, $\overleftarrow{2}\overleftarrow{1}\overrightarrow{3}$

Επίδοση

Δημιουργία μεταθέσεων

Ο αλγόριθμος Johnson-Trotter δεν παράγει μεταθέσεις σε λεξικογραφική διάταξη

Παράδειγμα 123, 132, 213, 231, 312, 321

Λύση:

- Σαρώνουμε από δεξιά προς τα αριστερά μέχρι να βρούμε ένα ζεύγος (a_i, a_{i+1}) που να ισχύει $a_i < a_{i+1}$.
- Κατόπιν, βρίσκουμε το μικρότερο στοιχείο στις θέσεις *i+1,…n* που είναι μεγαλύτερο του a_i και το τοποθετούμε στη θέση *i.*
- Τα υπόλοιπα στοιχεία δεξιά από αυτή τη θέση μπαίνουν σε αύξουσα σειρά.

Δημιουργία υποσυνόλων (1)

Χρήσιμη στο πρόβλημα του σάκου (εξαντλητική λύση).

Δυναμοσύνολο : ένα σύνολο 2^n συνόλων

Μείωση κατά ένα

Ιδέα: δημιουργούμε ένα δυναμοσύνολο με τα αντικείμενα {1,2,…, *n*-1} και εισάγουμε το αντικείμενο *n* στο καθένα από αυτά.

Δημιουργία υποσυνόλων (2)

Χρησιμοποιούμε ένα bitstring για την αναπαράσταση των συνόλων: 101 ≈ $\{a_1, a_3\}$

Παράδειγμα: για 3 αντικείμενα έχουμε 8 σύνολα που αναπαρίστανται με 000, 001, 010, …, 110, 111.

Επεκτάσεις

- Ταυτόχρονα λεξικογραφική διάταξη, συμπιεσμένη διάταξη (squashed order)
- Διάταξη με ελάχιστη αλλαγή (minimal change order)

  Gray code: 000, 001, 011, 010, 110, 111, 101, 100

Μείωση κατά σταθερό όρο: κίβδηλο νόμισμα

- Μεταξύ *n* νομισμάτων με ίδια όψη, το ένα είναι κίβδηλο. Έστω ότι είναι ελαφρύτερο.
- Χωρίζουμε τα νομίσματα σε δύο κατηγορίες από $\lfloor n/2 \rfloor$ νομίσματα, συν 1 νόμισμα αν το *n* είναι περιττός αριθμός
- Αναδρομική εξίσωση:
- Τι θα συμβεί αν χωρίσουμε σε 3 κατηγορίες;

Πολλαπλασιασμός αλά ρωσικά

- Αλλιώς καλείται μέθοδος του ρώσου χωρικού
- Έστω δύο θετικοί ακέραιοι n και m
- Αν το n είναι άρτιο, τότε : $n \cdot m$ = n/2 $\cdot$ 2m
- Αν το n είναι περιττό, τότε : $n \cdot m = (n\text{-}1)/2 \cdot 2m + m$

Παράδειγμα : 20 * 26

n	m	
20	26	
10	52	
5	104	104
2	208	+
1	416	416
		= **520**

Το πρόβλημα του Josephus

- Ιστορικό πρόβλημα
- *N* αποκλεισμένοι άνδρες , για να μην παραδοθούν, θα σκοτώσουν διαδοχικά ο ένας τον άλλο. Να βρεθεί ο τελευταίος επιζών.
- Παράδειγμα για *N* = 6 , 7
- Αναδρομική εξίσωση για τη θέση του επιζώντα
  – J(2k) = 2J(k)-1
  – J(2k+1) = 2J(k)+1

Μείωση κατά μεταβλητό μέγεθος: υπολογισμός μέσου

- Στατιστικά διατάξεων: βρες τον *k*-οστό αριθμό
- Μία λύση είναι να ταξινομήσουμε και να επιλέξουμε τον *k*-οστό
- Καλύτερα να διαμερίσουμε όπως στη γρήγορη ταξινόμηση
- Παράδειγμα : 4, 1, 10, 9, 7, 12, 8, 2, 15
- Επίδοση στην καλύτερη, χειρότερη και μέση περίπτωση

Παράδειγμα εύρεσης μέσου

Παράδειγμα: 4 1 10 9 7 12 8 2 15 $n = 9, k = \lceil 9/2 \rceil = 5$

1	2	3	4	5	6	7	8	9	
4	1	*10*	9	7	12	8	*2*	15	
4	1	2	9	7	12	8	10	15	
2	1	4	9	7	12	8	10	15	s=3<k=5
			9	7	*12*	*8*	10	15	
			9	7	8	12	10	15	
			8	7	9	12	10	15	s=6>k=5
	8	7							
	7	8							s=k=5

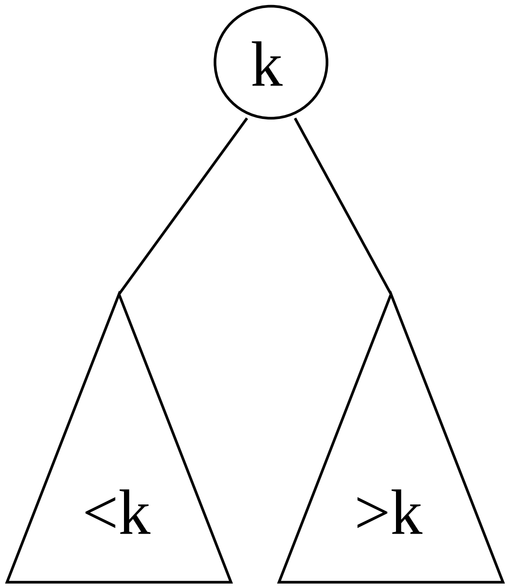
Λύση: median = 8

Αναζήτηση παρεμβολής

- Η δυαδική αναζήτηση δεν είναι πανάκεια
- Δοθέντος ενός ταξινομημένου πίνακα A, όπου το A[l] είναι το ελάχιστο και A[r] το μέγιστο, να βρεθεί η θέση της τιμής v.
- Λύση : δοκίμασε τη θέση
  $x = l + \lfloor (v-A[l])(r-l)/(A[r]-A[l]) \rfloor$
- Επίδοση της καλύτερης, χειρότερης και μέσης περίπτωσης: O(1), O(log logn), O(n)

Δυαδικά δένδρα αναζήτησης

- Τοποθέτηση κλειδιών σε ένα δυαδικό δένδρο με την ιδιότητα του δυαδικού δένδρου αναζήτησης:



Παράδειγμα 1: 5, 10, 3, 1, 7, 12, 9

Παράδειγμα 2: 4, 5, 7, 2, 1, 3, 6

Αναζήτηση και εισαγωγή σε δυαδικό δένδρο

- Αναζήτηση – εύκολη δουλειά
- Εισαγωγή – αναζήτηση κλειδιού και εισαγωγή στο φύλλο όπου καταλήγει η αναζήτηση
- Στη χειρότερη περίπτωση: h+1 συγκρίσεις κλειδιών
- $\lfloor \lg n \rfloor \leq h \leq n-1$
- Κατά μέσο όρο για τυχαία αρχεία: $h = 1.41 \lg n$
- Άρα:
  - Χειρότερη περίπτωση: $\Theta(n)$
  - Μέση περίπτωση: $\Theta(\lg n)$
- <u>Πλεονέκτημα:</u> η ενδοδιατεταγμένη διάσχιση παράγει ταξινομημένες λίστες

Σημείωμα Αναφοράς

Σημείωμα Αδειοδότησης

Τέλος ενότητας

Επεξεργασία: Ανδρέας Κοσματόπουλος
Θεσσαλονίκη, Αύγουστος 2015

Ευρωπαϊκή Ένωση
Ευρωπαϊκό Κοινωνικό Ταμείο
ΕΠΙΧΕΙΡΗΣΙΑΚΟ ΠΡΟΓΡΑΜΜΑ
ΕΚΠΑΙΔΕΥΣΗ ΚΑΙ ΔΙΑ ΒΙΟΥ ΜΑΘΗΣΗ
επένδυση στην κοινωνία της γνώσης
ΥΠΟΥΡΓΕΙΟ ΠΑΙΔΕΙΑΣ ΚΑΙ ΘΡΗΣΚΕΥΜΑΤΩΝ
ΕΙΔΙΚΗ ΥΠΗΡΕΣΙΑ ΔΙΑΧΕΙΡΙΣΗΣ
ΕΣΠΑ 2007-2013
πρόγραμμα για την ανάπτυξη
ΕΥΡΩΠΑΪΚΟ ΚΟΙΝΩΝΙΚΟ ΤΑΜΕΙΟ
Με τη συγχρηματοδότηση της Ελλάδας και της Ευρωπαϊκής Ένωσης

Σημειώματα

Σημείωμα Ιστορικού Εκδόσεων Έργου

Το παρόν έργο αποτελεί την έκδοση 1.00.

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.