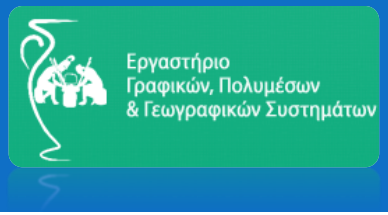
Αναζήτηση - Searching

Το πρόβλημα του Λεξικού (Κεφ. 4)

Δομές Δεδομένων

Μάριος Κενδέα



3 Μαρτίου 2015

kendea@ceid.upatras.gr

Περιεχόμενα

Εισαγωγή (1/3)

- Έστω σύμπαν U πηγή ενδιαφέροντος
- Σύνολο $S \subseteq U$, $x \in S$ και $\text{info}(x)$
- Λεξικό (Dictionary) υποστηρίζει πράξεις:
  - Access(x): if $x \in S$ true{return info(x)} else false — Στατικές Δομές
  - Insert(x): $S \rightarrow S \cup \{x\}$
  - Delete(x): $S \rightarrow S - \{x\}$ — Δυναμικές Δομές

Εισαγωγή (2/3)

- Insert(x) και Delete(x) είναι καταστρεπτικές πράξεις
  - Εφήμερες Δομές: κάθε πράξη ενημέρωσης αλλάζει την δομή
- Διαχρονικές Δομές (Persistent): Δομές που υποστηρίζουν Insert(x) και Delete(x) με Μη-Καταστρεπτικό τρόπο και διατηρούν όλες τις εκδόσεις
  - Μόνο Access -> Μερικώς διαχρονικές
  - Insert, Delete σε προηγούμενες εκδόσεις -> Πλήρως διαχρονικές
- Κατηγοριοποίηση ως προς το χώρο
  - **Συνοπτικές Δομές**: ουρά με χρήση πίνακα -> n+O(1) χώρο
  - Εκτενείς Δομές: ουρά με χρήση δεικτών -> O(n) χώρο

Εισαγωγή (3/3)

- Πως κάνουμε αναζήτηση σε ένα λεξικό και πως επιλέγουμε κάθε φορά το επόμενο σημείο αν δεν έχουμε απάντηση?
  - Γραμμικό ψάξιμο (Linear Search)
    - $next \leftarrow \text{left} + 1$
    - O(n)
  - Δυαδικό ψάξιμο (binary search)
  - Ψάξιμο παρεμβολής (interpolation search)

Δυαδική Αναζήτηση

- $next \leftarrow \left\lfloor \frac{right+left}{2} \right\rfloor$
- Βρίσκουμε το μέσο του διαστήματος
- Πάμε στο υποδιάστημα που μας ενδιαφέρει
- Loop until
  - Μήκος υποδιαστήματος = 1 -> Λύση
  - Μήκος υποδιαστήματος = 0 -> Δεν Υπάρχει
- Χειρότερη περίπτωση χρόνος **O(log n)**

Αναζήτηση Παρεμβολής (1/2)

- Αναζήτηση ανάλογη της αίσθησης που έχει ο άνθρωπος όταν αναζητά σε ένα λεξικό.
- $next \leftarrow \left\lfloor \frac{x - S[left]}{S[right] - S[left]} (right - left) \right\rfloor + left$
- Πόσο μεγαλύτερο είναι το ζητούμενο στοιχείο από το αριστερό άκρο
- Πόσο μεγαλύτερο είναι το δεξί άκρο από το αριστερό
- Παίζει ρόλο στην απόδοση το πόσο διαφέρουν τα στοιχεία από τα γειτονικά
- Χειρότερη περίπτωση χρόνος O(n)
- Μέσος χρόνος O(log logn) -> Υπερισχύει της δυαδικής αναζήτησης

Αναζήτηση Παρεμβολής (2/2)

- Πρόβλημα Απόδοσης

Δυαδική Αναζήτηση Παρεμβολής (1/3)

- Χειρότερη περίπτωση χρόνος O($\sqrt{n}$)
- Μέσος χρόνος O(log logn)
- Κάνουμε γραμμική επανάληψη με βήμα $\sqrt{n}$

- Ακολουθεί ο αλγόριθμος
  - Προσοχή είναι ψευδοκώδικας! Δεν είναι έτοιμος για υλοποίηση
    - Θεωρεί 1^{η} θέση το 1 όχι το 0 και τελευταία το n όχι το n-1
    - Στη γραμμή 8 μικρά διαστήματα κάνει γραμμική αναζήτηση
    - Δεν χειρίζεται ακραίες περιπτώσεις

Δυαδική Αναζήτηση Παρεμβολής (2/3)

Δυαδική Αναζήτηση Παρεμβολής (3/3) Βελτίωση

- **Βελτίωση χρόνου χειρότερης περίπτωσης** σε O(logn) το βήμα στο loop από i=i+1 γίνεται **i=2*i**
- 1ο βήμα: Εκθετικά βήματα εξετάζοντας τα διαστήματα [next, next+$\sqrt{n}$], [next, next+$2\sqrt{n}$], [next, next+$2^2\sqrt{n}$], [next, next+$2^3\sqrt{n}$],…, [next, next+$2^{\mathrm{i}}\sqrt{n}$] μέχρι να βρεθεί το διάστημα που περιέχει το x: S[next+$2^{\mathrm{j}-1}\sqrt{n}$] $< x \leq$ [next+$2^j\sqrt{n}$]
  - Άρα κάνουμε εκθετική επανάληψη: $\sqrt{n}, 2\sqrt{n}, 4\sqrt{n}, 8\sqrt{n}, \ldots$
- 2ο βήμα: Ψάχνουμε **δυαδικά στα στοιχεία που απέχουν $\sqrt{n}$** στο διάστημα του 1ου βήματος και προκύπτει το διάστημα $\sqrt{n}$ που περιέχει το στοιχείο.

?