Δομές Δεδομένων και Αλγόριθμοι - Εισαγωγή

Στην ενότητα αυτή θα μελετηθούν τα εξής επιμέρους θέματα:

Εισαγωγή στις έννοιες Αλγόριθμοι και Πολυπλοκότητα, Οργάνωση Δεδομένων και Δομές Δεδομένων

Χρήσιμοι μαθηματικοί ορισμοί

Μαθηματική Επαγωγή

Δεδομένα

- Τι είναι τα *Δεδομένα*;

  Αφαιρετική όψη της πραγματικότητας.

  Από τα δεδομένα προκύπτει η πληροφορία.

- Τα δεδομένα που χρησιμοποιούμε δεν έχουν πάντοτε τον ίδιο βαθμό πολυπλοκότητας. Μερικά μπορούν να αναλυθούν σε απλούστερα συστατικά, ενώ άλλα δεν μπορούν. Δεδομένα που δεν μπορούν να αναλυθούν λέγονται *πρωτογενή δεδομένα*.

- Τα δεδομένα ενός προβλήματος συνήθως δεν είναι μια άμορφη συλλογή στοιχείων. Τις περισσότερες φορές βρίσκονται σε κάποια σχέση το ένα προς το άλλο, π.χ.
  - σαν ένα απλό σύνολο: $\{2, 3, 5, 7, 11, 13\}$
  - σαν ένα διάνυσμα: $(2, 3, 5, 7, 11, 13)$
  - σαν ένας πίνακας: $\begin{pmatrix} 2 & 3 & 5 \\ 7 & 11 & 13 \end{pmatrix}$

Δεδομένα

- Οι *δομές δεδομένων* είναι συλλογές πρωτογενών δεδομένων, που συνδυάζονται για να σχηματίσουν πολυπλοκότερα δεδομένα.
- Θα χρησιμοποιήσουμε τον όρο *πίνακας* για να περιγράψουμε μια συλλογή στοιχείων τα οποία θα ονομάσουμε *κόμβους* ή *καταχωρήματα.*
- Ένας κόμβος μπορεί να είναι *απλός*, δηλαδή να αποτελείται από ένα μόνο πρωτογενές δεδομένο, ή να είναι *σύνθετος*, οπότε αποτελείται από δύο ή περισσότερα *πεδία*. Τα πεδία ενός κόμβου μπορεί να αντιπροσωπεύουν πρωτογενή δεδομένα διαφόρων τύπων.

Δεδομένα

- Ένα πεδίο ή ένα καταχώρημα χαρακτηρίζεται από τη *διεύθυνση* του και το *μήκος* του. Η διεύθυνση ενός πεδίου ή ενός καταχωρήματος είναι η διεύθυνση της πρώτης του κυψελίδας μνήμης και το μήκος του είναι ο αριθμός κυψελίδων από τις οποίες αποτελείται.

	← 1 byte →	← 4 bytes →
1040	A	1960
1045	True	72418
1050		

- Η διεύθυνση του πρώτου πεδίου, του πρώτου καταχωρήματος και ολόκληρου του πίνακα είναι 1040. Η διεύθυνση του δεύτερου καταχωρήματος 1050, και ούτω καθεξής.

Ενδόμνημη Παράσταση Δομών Δεδομένων

- Για να αναφερθούμε σε ένα καταχώρημα ή σε ένα πεδίο, χρησιμοποιούμε συνήθως συμβολικά ονόματα (όπως σε γλώσσες προγραμματισμού η ονομασία μεταβλητών αντιστοιχεί σε μια λέξη της μνήμης).

- Για να πραγματοποιηθεί μια δομή δεδομένων, απαιτείται η συσχέτιση των συμβολικών ονομάτων με αντίστοιχα τμήματα της μνήμης του υπολογιστή. Αυτό μπορεί να γίνει με δύο τρόπους:
  - *Διαδοχική Χορήγηση Μνήμης*

    Παράδειγμα: Αποθήκευση πίνακα:

    1. αποθήκευση κατά γραμμές
    2. αποθήκευση κατά στήλες

    Αποθήκευση πίνακα ($m \times n$) κατά στήλες μπορεί να γίνει με βάση τη *συνάρτηση απεικόνισης*:

    $loc(a[i,j]) = loc(a[1,1]) + (i-1) + m*(j-1)$

Ενδόμνημη Παράσταση Δομών Δεδομένων

– Συνδετική Χορήγηση Μνήμης

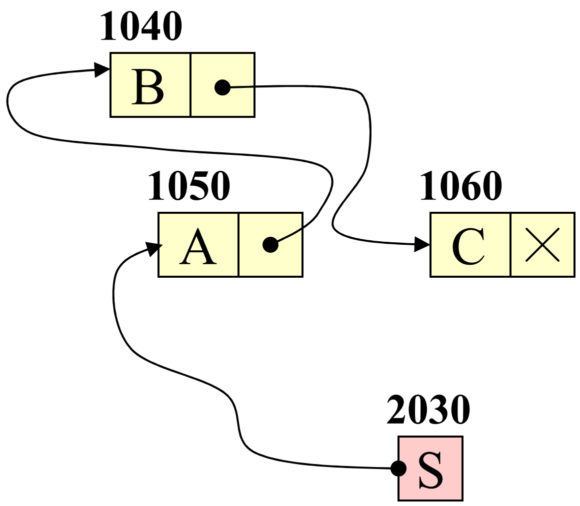
Ένα πεδίο μπορεί να παριστά τη διεύθυνση ενός άλλου πεδίου. Έτσι κατά τη συνδετική χορήγηση μνήμης κάθε κόμβος περιέχει πληροφορία σχετικά με το που βρίσκεται ο επόμενος κόμβος της δομής. Για την αποθήκευση αυτής της πληροφορίας κάθε κόμβος χρειάζεται ένα ειδικό πεδίο. Το πεδίο αυτό είναι χαρακτηριστικό της συνδετικής χορήγησης μνήμης.

Παράδειγμα: Συνδεδεμένη Λίστα

Παράδειγμα: Συνδεδεμένη Λίστα

- Οι κόμβοι της λίστας αποτελούνται από δύο πεδία: Info, για καταχώρηση πληροφοριών (των στοιχείων της λίστας), και Next (τύπου pointer) για αποθήκευση του δείκτη που δείχνει τη διεύθυνση του επόμενου κόμβου της λίστας.

	Info	Next
1040	B	1060
1045		
1050	A	1040
1065		
1060	C	NIL
. . .	. . .	. . .
2030	S	1050



Αλγόριθμοι

Abu Jafar Mohammed ibn Musa al Khowarizmi

Τι είναι ένας αλγόριθμος;

Αλγόριθμος είναι μια πεπερασμένη ακολουθία εντολών, αυστηρά καθορισμένων και εκτελέσιμων σε πεπερασμένο χρόνο, οι οποίες αν ακολουθηθούν επιτυγχάνεται κάποιο επιθυμητό αποτέλεσμα.

Απαραίτητα κριτήρια:

- Υπάρχει είσοδος και έξοδος
- Καθορισμός εντολών (όχι ασάφειες).
- Περατότητα.

Δηλαδή, δοθέντος ενός προβλήματος, ένας αλγόριθμος παρέχει τις οδηγίες σύμφωνα με τις οποίες τα *δεδομένα* του προβλήματος *μετασχηματίζονται* και *συνδυάζονται* για να προκύψει η λύση του προβλήματος.

Αλγόριθμοι

- Κατά την εκτέλεση ενός αλγόριθμου η δομή που έχουν τα δεδομένα παίζει πολύ μεγάλο ρόλο.

- Εξίσωση Wirth

  Αλγόριθμοι + Δεδομένα = Προγράμματα

Κεντρικός στόχος του μαθήματος είναι η μελέτη δομών δεδομένων, αναπαράστασής τους στη μνήμη ενός υπολογιστή, και αλγόριθμων που τις δημιουργούν και τις επεξεργάζονται.

Το πρόβλημα επιλογής

- Πρόβλημα: Έστω ότι έχουμε *n* αριθμούς και θέλουμε να προσδιορίσουμε τον *k*-οστό πιο μεγάλο.

  π.χ. Έστω οι αριθμοί 5, 72, 3, 4, 1, 9, 65.

  Ο δεύτερος πιο μεγάλος είναι ο 3, ο έκτος πιο μεγάλος είναι ο 65, κοκ.

- Αυτό το πρόβλημα είναι γνωστό ως το πρόβλημα επιλογής (the selection problem). Υπάρχουν διάφοροι 'εύκολοι' τρόποι λύσης:
  1. 'Διαβάζουμε' τους *n* αριθμούς σε μια λίστα, *a*. Ταξινομούμε τη λίστα από το μικρότερο στο μεγαλύτερο με βάση κάποιο αλγόριθμο ταξινόμησης. Επιστρέφουμε το *k*-οστό στοιχείο της λίστας, δηλαδή το *a*[*k*].
  2. 'Διαβάζουμε' τους *k* πρώτους αριθμούς σε μια λίστα *a*. Ταξινομούμε τη λίστα από το μικρότερο στο μεγαλύτερο. Μετά, επεξεργαζόμαστε τους υπόλοιπους *n* - *k* αριθμούς ως εξής: αν ένα στοιχείο είναι πιο μεγάλο από το *a*[*k*] το αγνοούμε, διαφορετικά το τοποθετούμε στη σωστή θέση στη λίστα. Όταν η διαδικασία αυτή τελειώσει, επιστρέφουμε το *k*-οστό στοιχείο της λίστας, δηλαδή το *a*[*k*].

Το πρόβλημα επιλογής

- Ποιος από τους δύο αλγόριθμους είναι ο καλύτερος;
- Οι αλγόριθμοι είναι ικανοποιητικοί;
- Υλοποίηση των αλγορίθμων και εφαρμογή τους σε υπολογιστή με n=1,000,000 και k=500,000 χρειάζεται πάρα πολύ χρόνο για να τερματίσει.
- Εντούτοις υπάρχει αλγόριθμος που επιτυγχάνει το ίδιο αποτέλεσμα σε δευτερόλεπτα.

Αλγόριθμοι

- Σημαντικό συμπέρασμα είναι πως το να γράφουμε ένα σωστό πρόγραμμα δεν είναι αρκετό. Ιδιαίτερα, όταν το σύνολο αρχικών δεδομένων είναι μεγάλου μεγέθους, ο *χρόνος εκτέλεσης* ενός προγράμματος είναι κύριας σημασίας.

- Στο μάθημα αυτό θα μάθουμε να υπολογίζουμε το χρόνο εκτέλεσης αλγόριθμων και να συγκρίνουμε την αποδοτικότητα διαφορετικών αλγόριθμων, προτού τους υλοποιήσουμε. Θα μελετήσουμε επίσης μεθόδους βελτίωσης της ταχύτητας προγραμμάτων.

- Πρόβλημα: Εισηγηθείτε μέθοδο σύμφωνα με την οποία μπορείτε να υλοποιήσετε πρόγραμμα το οποίο
  - διατηρεί ένα σύνολο από προσωπικά στοιχεία φίλων σας (διεύθυνση, τηλέφωνο, κλπ) και
  - τα επεξεργάζεται (συμπληρώνει, μεταλλάσσει, κλπ).

Ανασκόπηση Μαθηματικών Ορισμών

Ορισμός 1

Λέγοντας *σύνολο* εννοούμε μια ολότητα οντοτήτων. Τις οντότητες αυτές ονομάζουμε *στοιχεία* του συνόλου. π.χ. {True, False}

Ορισμός 2

Δοθέντων δύο στοιχείων a και b, *διατεταγμένο ζεύγος* (a,b) ονομάζουμε το σύνολο {{a},{a,b}}, όπου a είναι το πρώτο μέρος του ζεύγους και b το δεύτερο μέρος του ζεύγους.

Ορισμός 3

Δοθέντων δύο συνόλων A και B *καρτεσιανό γινόμενο* A×B ονομάζουμε το σύνολο όλων των διατεταγμένων ζευγών που έχουν ως πρώτο μέλος τους στοιχείο από το σύνολο A και ως δεύτερο μέλος τους στοιχείο από το σύνολο B, δηλαδή

- $A \times B = \{(a,b) \mid a \in A$ και $b \in B\}$
- Αν το σύνολο A έχει n στοιχεία και το σύνολο B έχει m στοιχεία, τότε τα σύνολα A×B και B×A έχουν $n \cdot m$ στοιχεία.

Μαθηματικοί Ορισμοί

Ορισμός 4

Δυαδική σχέση των συνόλων Α και Β ονομάζουμε οποιοδήποτε υποσύνολο του συνόλου Α×Β. Αν Σ είναι μια σχέση, για να εκφράσουμε πως το α και το β σχετίζονται με βάση αυτή τη σχέση, γράφουμε $(\alpha,\beta)\in\Sigma$ ή $\alpha\ \Sigma\ \beta$.

Ορισμός 5

Αν Σ είναι μια σχέση εντός ενός συνόλου Α, τότε η σχέση λέγεται:

- *ανακλαστική* (ή αυτοπαθής), αν για κάθε $\alpha \in A$, $\alpha\Sigma\alpha$
- *συμμετρική*, αν για κάθε $\alpha, \beta \in A$, $\alpha\ \Sigma\ \beta \Rightarrow \beta\ \Sigma\ \alpha$
- *μεταβατική*, αν για κάθε $\alpha, \beta, \gamma \in A$, $\alpha\ \Sigma\ \beta$ και $\beta\ \Sigma\ \gamma \Rightarrow \alpha\ \Sigma\ \gamma$

Ορισμός 6

Μια σχέση εντός ενός συνόλου Α λέγεται *σχέση ισοδυναμίας*, αν είναι ανακλαστική, συμμετρική και μεταβατική.

Λογάριθμοι, ισοτιμίες, κλπ

<u>Ορισμός 1:</u> $\log_x a = b$ αν $x^b = a$

<u>Χρήσιμοι νόμοι λογάριθμων:</u>

$$\log ab = \log a + \log b \qquad \log(a/b) = \log a - \log b$$

$$\log_a b = \frac{\log_c b}{\log_c a} \qquad \log a^b = b \cdot \log a$$

<u>Ορισμός 2:</u> Λέμε πως

το a είναι *ισότιμο* με το b ως προς μέτρο n, a ≡ b mod n,

αν το n διαιρεί το a-b.

π.χ. 8 ≡ 3 mod 5, 9 ≡ 0 mod 3.

<u>Ορισμός 3:</u> $\lfloor x \rfloor = \max\{a \mid a \le x, \mathrm{int}(a)\}$

$$\lceil x \rceil = \min\{a \mid a \ge x, \mathrm{int}(a)\}$$

π.χ. $\left\lceil \frac{63}{11} \right\rceil = 6, \quad \lfloor 5.1634 \rfloor = 5$

Σειρές

$$\sum_{i=0}^{n} i = \frac{n \cdot (n+1)}{2}$$

$$\sum_{i=0}^{n} i^2 = \frac{n \cdot (n+1)(2n+1)}{6}$$

$$\sum_{i=0}^{n} 2^i = 2^{n+1} - 1$$

$$\sum_{i=0}^{n} a^i = \frac{a^{n+1} - 1}{a - 1}$$

$$n + n/2 + n/4 + \ldots + 2 + 1 = 2n - 1$$

Μαθηματική Επαγωγή

Στόχος

Να αποδειχτεί ότι η (μαθηματική) πρόταση Π(n) ισχύει για κάθε $n \geq 0$.

Μέθοδος

1. Επαληθεύουμε πως η Π ισχύει για $n=0$,
2. Υποθέτουμε πως η Π ισχύει για $n=k$ και
3. Αποδεικνύουμε πως η Π ισχύει για $n=k+1$.

Παραλλαγές

- Αντί του 0, σε ορισμένες περιπτώσεις ενδιαφερόμαστε για $n \geq a$, όπου το a είναι κάποιος ακέραιος.
- Στο δεύτερο βήμα: Υποθέτουμε πως η Π ισχύει για $n \leq k$ και αποδεικνύουμε πως η Π ισχύει για $n=k+1$.

Παράδειγμα

Να αποδείξετε ότι $\sum_{i=0}^{n} i = \frac{n \cdot (n+1)}{2}$ για κάθε n ≥ 0.

Απόδειξη:

Θα αποδείξουμε με τη μέθοδο της μαθηματικής επαγωγής την πρόταση

$$\Pi(n) \equiv \sum_{i=0}^{n} i = \frac{n \cdot (n+1)}{2} .$$

1. Προφανώς η Π(0) ισχύει αφού $\sum_{i=0}^{0} i = 0 = \frac{0 \cdot (0+1)}{2}$.
2. Υποθέτουμε ότι ισχύει η Π(k), δηλαδή ότι $\sum_{i=0}^{k} i = \frac{k \cdot (k+1)}{2}$.
3. Και θα αποδείξουμε ότι ισχύει η Π(k+1):

$$\sum_{i=0}^{k+1} i = \sum_{i=0}^{k} i + (k+1) = \frac{k \cdot (k+1)}{2} + (k+1)$$

$$= \frac{(k+1)(k+2)}{2} \qquad \square$$

Εργασία

- Αποδείξετε με τη μέθοδο μαθηματικής επαγωγής τις σειρές της διαφάνειας 1-16

- Αποδείξετε με τη μέθοδο μαθηματικής επαγωγής ότι για κάθε $n \geq 7$,

$$3^n < n!$$