Μαθηματικά Πληροφορικής

6ο Μάθημα

Ηλίας Κουτσουπιάς

Τμήμα Πληροφορικής και Τηλεπικοινωνιών
Πανεπιστήμιο Αθηνών
elias@@di.uoa.gr

Γενικό πλάνο

1. Ανάλυση αλγορίθμων
2. Συμβολισμοί O, Ω και Θ
3. Αναδρομικές εξισώσεις

Είδη αλγορίθμων

- Αλγόριθμοι με αρχή και τέλος, με είσοδο και έξοδο

Είδη αλγορίθμων

- Αλγόριθμοι με αρχή και τέλος, με είσοδο και έξοδο
- Ατέρμονες αλγόριθμοι (λειτουργικά συστήματα, δομές δεδομένων, πρωτόκολλα Διαδικτύου).

Είδη αλγορίθμων

- Αλγόριθμοι με αρχή και τέλος, με είσοδο και έξοδο
- Ατέρμονες αλγόριθμοι (λειτουργικά συστήματα, δομές δεδομένων, πρωτόκολλα Διαδικτύου).
- Παράλληλοι, κατανεμημένοι κλπ

Εδώ θα μελετήσουμε μόνο το πρώτο είδος και ειδικά το χρόνο εκτέλεσης των αλγορίθμων.

Υποθέσεις και συμβάσεις

- Ο χρόνος εκτέλεσης ενός αλγορίθμου (προγράμματος) εξαρτάται από

Υποθέσεις και συμβάσεις

- Ο χρόνος εκτέλεσης ενός αλγορίθμου (προγράμματος) εξαρτάται από
  - την είσοδο

Υποθέσεις και συμβάσεις

- Ο χρόνος εκτέλεσης ενός αλγορίθμου (προγράμματος) εξαρτάται από
  - την είσοδο
  - τη γλώσσα προγραμματισμού

Υποθέσεις και συμβάσεις

- Ο χρόνος εκτέλεσης ενός αλγορίθμου (προγράμματος) εξαρτάται από
  - την είσοδο
  - τη γλώσσα προγραμματισμού
  - τον επεξεργαστή

Υποθέσεις και συμβάσεις

- Ο χρόνος εκτέλεσης ενός αλγορίθμου (προγράμματος) εξαρτάται από
    - την είσοδο
    - τη γλώσσα προγραμματισμού
    - τον επεξεργαστή
    - το υπολογιστικό περιβάλλον

Υποθέσεις και συμβάσεις

- Ο χρόνος εκτέλεσης ενός αλγορίθμου (προγράμματος) εξαρτάται από
    - την είσοδο
    - τη γλώσσα προγραμματισμού
    - τον επεξεργαστή
    - το υπολογιστικό περιβάλλον
    - ...

Υποθέσεις και συμβάσεις

- Ο χρόνος εκτέλεσης ενός αλγορίθμου (προγράμματος) εξαρτάται από
  - την είσοδο
  - τη γλώσσα προγραμματισμού
  - τον επεξεργαστή
  - το υπολογιστικό περιβάλλον
  - ...
- Για μια καλή θεωρία ανάλυσης αλγορίθμων πρέπει να

Υποθέσεις και συμβάσεις

- Ο χρόνος εκτέλεσης ενός αλγορίθμου (προγράμματος) εξαρτάται από
    - την είσοδο
    - τη γλώσσα προγραμματισμού
    - τον επεξεργαστή
    - το υπολογιστικό περιβάλλον
    - ...
- Για μια καλή θεωρία ανάλυσης αλγορίθμων πρέπει να
    - να ξεκαθαρίσουμε ποια χαρακτηριστικά της εισόδου επηρεάζουν το χρόνο του αλγορίθμου

Υποθέσεις και συμβάσεις

- Ο χρόνος εκτέλεσης ενός αλγορίθμου (προγράμματος) εξαρτάται από
  - την είσοδο
  - τη γλώσσα προγραμματισμού
  - τον επεξεργαστή
  - το υπολογιστικό περιβάλλον
  - ...
- Για μια καλή θεωρία ανάλυσης αλγορίθμων πρέπει να
  - να ξεκαθαρίσουμε ποια χαρακτηριστικά της εισόδου επηρεάζουν το χρόνο του αλγορίθμου
  - να εξαλείψουμε με κάποιο τρόπο τους υπόλοιπους εξωγενείς παράγοντες

Υποθέσεις και συμβάσεις

- Οι εξωγενείς παράγοντες επηρεάζουν κυρίως πολλαπλασιαστικά το χρόνο εκτέλεσης.

Υποθέσεις και συμβάσεις

- Οι εξωγενείς παράγοντες επηρεάζουν κυρίως πολλαπλασιαστικά το χρόνο εκτέλεσης.
  - Εκτέλεση σε υπολογιστή X: t

Υποθέσεις και συμβάσεις

- Οι εξωγενείς παράγοντες επηρεάζουν κυρίως πολλαπλασιαστικά το χρόνο εκτέλεσης.
  - Εκτέλεση σε υπολογιστή Χ: t
  - Εκτέλεση σε υπολογιστή Υ: αt

Υποθέσεις και συμβάσεις

- Οι εξωγενείς παράγοντες επηρεάζουν κυρίως πολλαπλασιαστικά το χρόνο εκτέλεσης.
  - Εκτέλεση σε υπολογιστή X: t
  - Εκτέλεση σε υπολογιστή Υ: αt
- Από τα χαρακτηριστικά της εισόδου, κρατάμε μόνο το μήκος της εισόδου.

Υποθέσεις και συμβάσεις

- Οι εξωγενείς παράγοντες επηρεάζουν κυρίως πολλαπλασιαστικά το χρόνο εκτέλεσης.
  - Εκτέλεση σε υπολογιστή X: t
  - Εκτέλεση σε υπολογιστή Υ: αt
- Από τα χαρακτηριστικά της εισόδου, κρατάμε μόνο το μήκος της εισόδου.

Βασική παραδοχή:

Ο χρόνος είναι συνάρτηση $T(n)$ του μήκους n της εισόδου.

Υποθέσεις και συμβάσεις

- Οι εξωγενείς παράγοντες επηρεάζουν κυρίως πολλαπλασιαστικά το χρόνο εκτέλεσης.
  - Εκτέλεση σε υπολογιστή Χ: t
  - Εκτέλεση σε υπολογιστή Υ: αt
- Από τα χαρακτηριστικά της εισόδου, κρατάμε μόνο το μήκος της εισόδου.

Βασική παραδοχή:

Ο χρόνος είναι συνάρτηση $T(n)$ του μήκους n της εισόδου.

- Ποιας εισόδου όμως αφού υπάρχουν 2^n είσοδοι μήκους n;

Υποθέσεις και συμβάσεις

- Οι εξωγενείς παράγοντες επηρεάζουν κυρίως πολλαπλασιαστικά το χρόνο εκτέλεσης.
  - Εκτέλεση σε υπολογιστή X: t
  - Εκτέλεση σε υπολογιστή Υ: αt
- Από τα χαρακτηριστικά της εισόδου, κρατάμε μόνο το μήκος της εισόδου.

Βασική παραδοχή:

Ο χρόνος είναι συνάρτηση $T(n)$ του μήκους n της εισόδου.

- Ποιας εισόδου όμως αφού υπάρχουν 2^n είσοδοι μήκους n;

Worst-case analysis:

$T(n)$ είναι ο μεγαλύτερος μεταξύ των χρόνων των εισόδων μήκους n.

Μήκος εισόδου

ΠΑΡΑΔΕΙΓΜΑ

- Είσοδος μια συμβολοσειρά του αλφαβήτου $\{0, 1\}$:
  Μήκος της εισόδου είναι ο αριθμός των συμβόλων της εισόδου.

Μήκος εισόδου

ΠΑΡΑΔΕΙΓΜΑ

- Είσοδος μια συμβολοσειρά του αλφαβήτου $\{0,1\}$:
  Μήκος της εισόδου είναι ο αριθμός των συμβόλων της εισόδου.
- Είσοδος μια συμβολοσειρά του αλφαβήτου $\{0,1,\ldots,9\}$:
  Μήκος εισόδου είναι 4 επί τον αριθμό των συμβόλων της εισόδου.

Μήκος εισόδου

ΠΑΡΑΔΕΙΓΜΑ

- Είσοδος μια συμβολοσειρά του αλφαβήτου $\{0,1\}$:
  Μήκος της εισόδου είναι ο αριθμός των συμβόλων της εισόδου.
- Είσοδος μια συμβολοσειρά του αλφαβήτου $\{0,1,\ldots,9\}$:
  Μήκος εισόδου είναι 4 επί τον αριθμό των συμβόλων της εισόδου.
- Είσοδος ένας φυσικός αριθμός a :
  Μήκος $\lceil \log a \rceil \approx \log a$.

Μήκος εισόδου

ΠΑΡΑΔΕΙΓΜΑ

- Είσοδος μια συμβολοσειρά του αλφαβήτου $\{0, 1\}$:
  Μήκος της εισόδου είναι ο αριθμός των συμβόλων της εισόδου.
- Είσοδος μια συμβολοσειρά του αλφαβήτου $\{0, 1, \ldots, 9\}$:
  Μήκος εισόδου είναι 4 επί τον αριθμό των συμβόλων της εισόδου.
- Είσοδος ένας φυσικός αριθμός a :
  Μήκος $\lceil \log a \rceil \approx \log a$.
- Είσοδος είναι μια ακολουθία $a_1, a_2, \ldots, a_k$ φυσικών αριθμών :
  Μήκος περίπου $\lceil \log a_1 \rceil + \lceil \log a_2 \rceil + \cdots + \lceil \log a_k \rceil$, αλλά πρέπει να λάβουμε υπόψη και τα κόμματα.

Μήκος εισόδου

ΠΑΡΑΔΕΙΓΜΑ

- Είσοδος μια συμβολοσειρά του αλφαβήτου $\{0,1\}$:
  Μήκος της εισόδου είναι ο αριθμός των συμβόλων της εισόδου.
- Είσοδος μια συμβολοσειρά του αλφαβήτου $\{0,1,\ldots,9\}$:
  Μήκος εισόδου είναι 4 επί τον αριθμό των συμβόλων της εισόδου.
- Είσοδος ένας φυσικός αριθμός a :
  Μήκος $\lceil \log a \rceil \approx \log a$.
- Είσοδος είναι μια ακολουθία $a_1, a_2, \ldots, a_k$ φυσικών αριθμών :
  Μήκος περίπου $\lceil \log a_1 \rceil + \lceil \log a_2 \rceil + \cdots + \lceil \log a_k \rceil$, αλλά πρέπει να λάβουμε υπόψη και τα κόμματα.

Είναι φανερό ότι ο καθορισμός του μήκους μιας εισόδου παρουσιάζει επιπλοκές και δυσκολίες.

Ορισμός

Έστω f και g δυο συναρτήσεις από τους μη αρνητικούς ακέραιους στο σύνολο των πραγματικών αριθμών. Θα λέμε ότι $f(n) = O(g(n))$ αν υπάρχουν θετικές σταθερές c και n_0 τέτοιες ώστε

$$f(n) \leq c \cdot g(n)$$

για κάθε $n \geq n_0$.

ΠΑΡΑΔΕΙΓΜΑ ($f(n) = 10n + 7$, $g(n) = n$, $10n + 7 = O(n)$)

Αρκεί να βρούμε τις δυο σταθερές c και n_0 του ορισμού και να επιβεβαιώσουμε τη σχέση $f(n) \leq c \cdot g(n)$.

$$n_0 = 1 \qquad c = 17$$

Για $n \geq n_0 = 1$, έχουμε

Συμβολισμός O

Πρόταση

Έστω $p(n)$ ένα πολυώνυμο βαθμού d. Τότε $p(n) = O(n^d)$.

Απόδειξη.

$$p(n) = a_d n^d + \cdots + a_1 n + a_0$$

Διαλέγουμε $n_0 = 1$ και $c = |a_d| + \cdots + |a_1| + |a_0|$ και έχουμε

$$\begin{aligned} p(n) &\leq |a_d n^d + \cdots + a_1 n + a_0| \\ &\leq |a_d n^d| + \cdots + |a_1 n| + |a_0| \\ &\leq |a_d| n^d + \cdots + |a_1| n + |a_0| \\ &\leq |a_d| n^d + \cdots + |a_1| n^d + |a_0| n^d \\ &= c n^d. \end{aligned}$$

Γιατί δεν διαλέξαμε $c = a_d + \cdots + a_1 + a_0$, χωρίς τις απόλυτες τιμές; $\square$

Τεχνικές

- Πως αποδεικνύουμε ότι $f(n) = O(g(n))$;

Τεχνικές

- Πως αποδεικνύουμε ότι $f(n) = O(g(n))$;
  - Βρίσκουμε κατάλληλα c και n_0.

Τεχνικές

- Πως αποδεικνύουμε ότι $f(n) = O(g(n))$;
  - Βρίσκουμε κατάλληλα c και n_0.
  - Για θετικές συναρτήσεις $f(n)$ και $g(n)$, δείχνουμε ότι το όριο

$$\lim_{n\to\infty} \frac{f(n)}{g(n)}$$

υπάρχει και είναι κάποια σταθερά (δεν είναι ∞ δηλαδή).

Τεχνικές

- Πως αποδεικνύουμε ότι $f(n) = O(g(n))$;
  - Βρίσκουμε κατάλληλα c και n_0.
  - Για θετικές συναρτήσεις $f(n)$ και $g(n)$, δείχνουμε ότι το όριο

$$\lim_{n\to\infty} \frac{f(n)}{g(n)}$$

    υπάρχει και είναι κάποια σταθερά (δεν είναι ∞ δηλαδή).
- Πως αποδεικνύουμε το αντίστροφο, δηλαδή ότι δεν ισχύει $f(n) = O(g(n))$;

Τεχνικές

- Πως αποδεικνύουμε ότι $f(n) = O(g(n))$;
  - Βρίσκουμε κατάλληλα c και n_0.
  - Για θετικές συναρτήσεις $f(n)$ και $g(n)$, δείχνουμε ότι το όριο

$$\lim_{n\to\infty} \frac{f(n)}{g(n)}$$

    υπάρχει και είναι κάποια σταθερά (δεν είναι ∞ δηλαδή).
- Πως αποδεικνύουμε το αντίστροφο, δηλαδή ότι δεν ισχύει $f(n) = O(g(n))$;
  - Δείχνουμε με απαγωγή σε άτοπο ότι δεν υπάρχουν κατάλληλα c και n_0.

Τεχνικές

- Πως αποδεικνύουμε ότι $f(n) = O(g(n))$;
  - Βρίσκουμε κατάλληλα c και n_0.
  - Για θετικές συναρτήσεις $f(n)$ και $g(n)$, δείχνουμε ότι το όριο

    $$\lim_{n\to\infty} \frac{f(n)}{g(n)}$$

    υπάρχει και είναι κάποια σταθερά (δεν είναι ∞ δηλαδή).
- Πως αποδεικνύουμε το αντίστροφο, δηλαδή ότι δεν ισχύει $f(n) = O(g(n))$;
  - Δείχνουμε με απαγωγή σε άτοπο ότι δεν υπάρχουν κατάλληλα c και n_0.
  - Για θετικές συναρτήσεις $f(n)$ και $g(n)$, δείχνουμε ότι το όριο

    $$\lim_{n\to\infty} \frac{f(n)}{g(n)}$$

    υπάρχει και είναι ∞.

Η τεχνική με τα όρια

Πρόταση

Για θετικές συναρτήσεις $f(n)$ *και* $g(n)$, *αν το όριο* $\lim_{n\to\infty} \frac{f(n)}{g(n)}$ *υπάρχει και είναι κάποια σταθερά (δεν είναι* ∞ *δηλαδή), τότε* $f(n) = O(g(n))$.

Απόδειξη.

Έστω a το όριο. Τι σημαίνει ότι $\lim_{n\to\infty} \frac{f(n)}{g(n)} = a$?
Ότι για κάθε $\epsilon > 0$, υπάρχει m_ϵ τέτοιο ώστε για κάθε $n \geq m_\epsilon$:

$$\left|\frac{f(n)}{g(n)} - a\right| \leq \epsilon$$

Ας πάρουμε $\epsilon = a$. Τότε για κάθε $n \geq m_a$:

$$\left|\frac{f(n)}{g(n)} - a\right| \leq a \quad \Rightarrow \quad \frac{f(n)}{g(n)} \leq 2a$$

Αρκεί τώρα να πάρουμε $n_0 = m_a$ και $c = 2a$. □

Παράδειγμα

Πρόταση

Να δειχτεί ότι δεν ισχύει $n^2 = O(n)$.

Απόδειξη.

Με εις άτοπο απαγωγή: Έστω ότι υπήρχαν κατάλληλες σταθερές c και n_0. Ας θεωρήσουμε την τιμή $n = \lceil \max\{n_0, c+1\} \rceil$. Θα πρέπει να έχουμε $n^2 \leq cn$. Ισοδύναμα $n \leq c$, άτοπο. □

Εναλλακτική απόδειξη.

Το

$$\lim_{n \to \infty} \frac{n^2}{n}$$

υπάρχει και είναι ∞. □

Σχέσεις του συμβολισμού O

Πρόταση

$A\nu\ f(n) = O(g(n))\ \kappa\alpha\iota\ g(n) = O(h(n))\ \tau o\tau\varepsilon\ f(n) = O(h(n))$

Πρόταση

$f_1(n) + f_2(n) = O(\max\{f_1(n), f_2(n)\})$

Ιεραρχία συναρτήσεων

Υπερεκθετικές:	2^{2^n}					
Εκθετικές:	2^{n^2}	n^n	$n!$	3^n	2^n	$2^{\sqrt{n}}$
Υποεκθετικές:	$n^{\log n}$					
Πολυωνυμικές:	n^2	$n \log n$	n	$\sqrt{n}$		
Λογαριθμικές:	$\log^2 n$	$\log n$				
Υπολογαριθμικές:	$\log \log n$					
Σταθερές	1					

Συμβολισμός Ω

Ορισμός

Έστω f και g δυο συναρτήσεις από τους μη αρνητικούς ακέραιους στο σύνολο των πραγματικών αριθμών. Θα λέμε ότι $f(n) = \Omega(g(n))$ αν υπάρχουν θετικές σταθερές c και n_0 τέτοιες ώστε

$$f(n) \geq c \cdot g(n)$$

για κάθε $n \geq n_0$.
Ισοδύναμα, $f(n) = \Omega(g(n))$ αν και μόνο αν $g(n) = O(f(n))$.

ΠΑΡΑΔΕΙΓΜΑ ($f(n) = 10n - 7$, $g(n) = n$, $10n - 7 = \Omega(n)$)

Αρκεί να βρούμε τις δυο σταθερές c και n_0 του ορισμού και να επιβεβαιώσουμε τη σχέση $f(n) \geq c \cdot g(n)$.

$$n_0 = 1 \qquad c = 3$$

$$f(n) = 10n - 7 \geq 10n - 7n \geq 3n = cn = cg(n).$$

Συμβολισμός Θ

Ορισμός

Έστω f και g δυο συναρτήσεις από τους μη αρνητικούς ακέραιους στο σύνολο των πραγματικών αριθμών. Θα λέμε ότι $f(n) = \Theta(g(n))$ αν $f(n) = O(g(n))$ και $f(n) = \Omega(g(n))$.

Πρόταση

Αν το $\lim_{n\to\infty} \frac{f(n)}{g(n)}$ *υπάρχει και έχει τιμή* λ *τότε*

$$f(n) = \begin{cases} O(g(n)) & \text{αν } \lambda = 0 \\ \Theta(g(n)) & \text{αν } 0 < \lambda < \infty \\ \Omega(g(n)) & \text{αν } \lambda = \infty \end{cases}$$

Βασικές αναδρομικές εξισώσεις

Στην ανάλυση αλγορίθμων χρειάζεται συχνά να βρούμε τη λύση ανδρομικών εξισώσεων. Τυπικές αναδρομικές εξισώσεις είναι οι εξής:

- $T(n) = T(n-1) + n$

Βασικές αναδρομικές εξισώσεις

Στην ανάλυση αλγορίθμων χρειάζεται συχνά να βρούμε τη λύση ανδρομικών εξισώσεων. Τυπικές αναδρομικές εξισώσεις είναι οι εξής:

- $T(n) = T(n-1) + n$
- $T(n) = 2T(\frac{n}{2}) + n$

Βασικές αναδρομικές εξισώσεις

Στην ανάλυση αλγορίθμων χρειάζεται συχνά να βρούμε τη λύση ανδρομικών εξισώσεων. Τυπικές αναδρομικές εξισώσεις είναι οι εξής:

- $T(n) = T(n-1) + n$
- $T(n) = 2T(\frac{n}{2}) + n$
- $T(n) = T(n-1) + T(n-2)$

Τεχνικές επίλυσης αναδρομικών

Για να λύσουμε μια αναδρομική χρησιμοποιούμε τρείς τεχνικές:

- Αν αποτελεί ειδική περίπτωση γνωστής οικογένειας αναδρομικών εξισώσεων, χρησιμοποιούμε τη γενικότερη λύση

Τεχνικές επίλυσης αναδρομικών

Για να λύσουμε μια αναδρομική χρησιμοποιούμε τρείς τεχνικές:

- Αν αποτελεί ειδική περίπτωση γνωστής οικογένειας αναδρομικών εξισώσεων, χρησιμοποιούμε τη γενικότερη λύση
- Μαντεύουμε τη λύση και την επιβεβαιώνουμε

Τεχνικές επίλυσης αναδρομικών

Για να λύσουμε μια αναδρομική χρησιμοποιούμε τρείς τεχνικές:

- Αν αποτελεί ειδική περίπτωση γνωστής οικογένειας αναδρομικών εξισώσεων, χρησιμοποιούμε τη γενικότερη λύση
- Μαντεύουμε τη λύση και την επιβεβαιώνουμε
- Την ‘ξεδιπλώνουμε’, τη γενικεύσουμε, και επιβεβαιώνουμε τη γενίκευση με χρήση επαγωγής.

Ξεδίπλωμα

- Ποια η λύση της αναδρομικής εξίσωσης $T(n) = T(n-1) + n$, με $T(1) = 1$;

Ξεδίπλωμα

- Ποια η λύση της αναδρομικής εξίσωσης $T(n) = T(n-1) + n$, με $T(1) = 1$;

$$\begin{aligned} T(n) &= T(n-1) + n \\ &= T(n-2) + (n-1) + n \\ &= T(n-3) + (n-2) + (n-1) + n \end{aligned}$$

Ξεδίπλωμα

- Ποια η λύση της αναδρομικής εξίσωσης $T(n) = T(n-1) + n$, με $T(1) = 1$;

$$\begin{aligned} T(n) &= T(n-1) + n \\ &= T(n-2) + (n-1) + n \\ &= T(n-3) + (n-2) + (n-1) + n \end{aligned}$$

- Μαντεύουμε τη γενίκευση $T(n) = T(n-k) + (n-k+1) + \cdots + n$, για κάθε $k \in \mathbb{N}$.

Ξεδίπλωμα

- Ποια η λύση της αναδρομικής εξίσωσης $T(n) = T(n-1) + n$, με $T(1) = 1$;

$$\begin{aligned} T(n) &= T(n-1) + n \\ &= T(n-2) + (n-1) + n \\ &= T(n-3) + (n-2) + (n-1) + n \end{aligned}$$

- Μαντεύουμε τη γενίκευση $T(n) = T(n-k) + (n-k+1) + \cdots + n$, για κάθε $k \in \mathbb{N}$.
- Αυτό το αποδεικνύουμε με επαγωγή στο k.

Ξεδίπλωμα

- Ποια η λύση της αναδρομικής εξίσωσης $T(n) = T(n-1) + n$, με $T(1) = 1$;

$$\begin{aligned} T(n) &= T(n-1) + n \\ &= T(n-2) + (n-1) + n \\ &= T(n-3) + (n-2) + (n-1) + n \end{aligned}$$

- Μαντεύουμε τη γενίκευση $T(n) = T(n-k) + (n-k+1) + \cdots + n$, για κάθε $k \in \mathbb{N}$.
- Αυτό το αποδεικνύουμε με επαγωγή στο k.
- Και τώρα για $k = n-1$, έχουμε

$$T(n) = T(1) + 2 + \cdots + n = 1 + 2 + \cdots + n = n(n+1)/2.$$

Ξεδίπλωμα

- Ποια η λύση της αναδρομικής εξίσωσης $T(n) = 3T(n-1) - 2T(n-2)$, με $T(1) = 1$ και $T(2) = 2$;

Ξεδίπλωμα

- Ποια η λύση της αναδρομικής εξίσωσης $T(n) = 3T(n-1) - 2T(n-2)$, με $T(1) = 1$ και $T(2) = 2$;

$$
\begin{aligned}
T(n) &= 3T(n-1) - 2T(n-2) \\
&= 7T(n-2) - 6T(n-3) \\
&= 15T(n-3) - 14T(n-4)
\end{aligned}
$$

Ξεδίπλωμα

- Ποια η λύση της αναδρομικής εξίσωσης $T(n) = 3T(n-1) - 2T(n-2)$, με $T(1) = 1$ και $T(2) = 2$;

$$\begin{aligned} T(n) &= 3T(n-1) - 2T(n-2) \\ &= 7T(n-2) - 6T(n-3) \\ &= 15T(n-3) - 14T(n-4) \end{aligned}$$

- Μαντεύουμε τη γενίκευση $T(n) = (2^k - 1)T(n-k+1) - (2^k - 2)T(n-k)$, για κάθε $k \geq 2$.

Ξεδίπλωμα

- Ποια η λύση της αναδρομικής εξίσωσης
  $T(n) = 3T(n-1) - 2T(n-2)$, με $T(1) = 1$ και $T(2) = 2$;

$$\begin{aligned} T(n) &= 3T(n-1) - 2T(n-2) \\ &= 7T(n-2) - 6T(n-3) \\ &= 15T(n-3) - 14T(n-4) \end{aligned}$$

- Μαντεύουμε τη γενίκευση
  $T(n) = (2^k - 1)T(n-k+1) - (2^k - 2)T(n-k)$, για κάθε $k \geq 2$.
- Αυτό το αποδεικνύουμε με επαγωγή στο k.

Ξεδίπλωμα

- Ποια η λύση της αναδρομικής εξίσωσης
  $T(n) = 3T(n-1) - 2T(n-2)$, με $T(1) = 1$ και $T(2) = 2$;

$$\begin{aligned} T(n) &= 3T(n-1) - 2T(n-2) \\ &= 7T(n-2) - 6T(n-3) \\ &= 15T(n-3) - 14T(n-4) \end{aligned}$$

- Μαντεύουμε τη γενίκευση
  $T(n) = (2^k - 1)T(n-k+1) - (2^k - 2)T(n-k)$, για κάθε $k \geq 2$.
- Αυτό το αποδεικνύουμε με επαγωγή στο k.
- Και τώρα για $k = n-1$, έχουμε

$$T(n) = (2^{n-1} - 1)T(2) - (2^{n-1} - 2)T(1) = 2^{n-1}.$$

Λύση ομογενών αναδρομικών εξισώσεων

Θεώρημα

Έστω $T(n) = a_1 T(n-1) + a_2 T(n-2) \cdots + a_r T(n-r)$ *μια αναδρομική εξίσωση. Αν οι μιγαδικές ρίζες* $x_1, x_2, \ldots, x_r$ *της πολυωνυμικής εξίσωσης* $x^r = a_1 x^{r-1} + a_2 x^{r-2} + \cdots + a_r$ *είναι διαφορετικές μεταξύ τους, η λύση της αναδρομικής εξίσωσης είναι της μορφής*

$$T(n) = c_1 x_1^n + c_2 x_2^n + \cdots + c_r x_r^n,$$

όπου $c_1, c_2, \ldots, c_n$ *σταθερές που καθορίζονται από τις αρχικές τιμές* $T(1), T(2), \ldots, T(r)$.

ΠΑΡΑΔΕΙΓΜΑ

$$T(n) = 3T(n-1) - 2T(n-2)$$

$$x^2 = 3x - 2 \quad \Rightarrow \quad x_1 = 2,\, x_2 = 1$$

$$T(n) = c_1 2^n + c_2 1^n = c_1 2^n + c_2$$