

Ανάλυση αλγορίθμων

Παράμετροι απόδοσης ενός αλγόριθμου:

- Χρόνος εκτέλεσης
- Απαιτούμενοι πόροι, π.χ. μνήμη, επικοινωνία (π.χ. σε κατανεμημένα συστήματα)
- Προσπάθεια υλοποίησης

Ανάλυση της απόδοσης

- Θεωρητική ανάλυση
- Πειραματική ανάλυση

Ανάλυση αλγορίθμων

Παράμετροι απόδοσης ενός αλγόριθμου:

- Χρόνος εκτέλεσης
- Απαιτούμενοι πόροι, π.χ. μνήμη, επικοινωνία (π.χ. σε κατανεμημένα συστήματα)
- Προσπάθεια υλοποίησης

Ανάλυση της απόδοσης

- Θεωρητική ανάλυση
- Πειραματική ανάλυση

Πότε μπορούμε να είμαστε
ευχαριστημένοι από την απόδοση
ενός αλγόριθμου;

Ανάλυση αλγορίθμων

Χρόνος εκτέλεσης:

- Αναμενόμενη περίπτωση
 - απαιτεί γνώση της κατανομής εισόδου
 - όταν η κατανομή δεν είναι γνωστή (κάτι που συμβαίνει συχνά!), συνήθως υποθέτουμε ομοιόμορφη κατανομή
 - πιθανοκρατικοί αλγόριθμοι διαμορφώνουν την κατανομή εισόδου
- Χειρότερη περίπτωση
 - ισχύει για κάθε είσοδο
 - κοντά στην αναμενόμενη όταν συμβαίνει συχνά, αλλά μπορεί να είναι απαισιόδοξη όταν συμβαίνει σπάνια

Ανάλυση αλγορίθμων

Πειραματική ανάλυση του χρόνου εκτέλεσης

```
public static void main()  
{  
    long startTime = System.currentTimeMillis();  
    // run algorithm  
    ...  
    long endTime    = System.currentTimeMillis();  
    long totalTime = endTime - startTime;  
}
```

Ανάλυση αλγορίθμων

Πειραματική ανάλυση του χρόνου εκτέλεσης

Χρόνοι εκτέλεσης σε δευτερόλεπτα για διάφορες δομές εύρεσης-ένωσης με είσοδο τυχαίο αρχείο με n κόμβους και m ακμές

n	m	UF	UFW	UFWP
100000	100000	2.090	0.124	0.140
100000	200000	19.172	0.249	0.264
100000	400000	63.429	0.546	0.530
100000	800000	153.692	1.075	1.045
400000	800000	378.645	1.247	1.138

UF = βασική δομή γρήγορης ένωσης

UFW = δομή γρήγορης ένωσης με στάθμιση

UFWP = δομή γρήγορης ένωσης με στάθμιση και συμπίεση διαδρομής

Ασυμπτωτική ανάλυση

Είναι η μελέτη του ρυθμού αύξησης του χρόνου εκτέλεσης (ή άλλης παραμέτρου απόδοσης) ως συνάρτηση του μεγέθους της εισόδου

Παράδειγμα: Υπολογισμός εσωτερικού γινομένου

$$z = x^T y = \sum_{i=1}^{i=n} x_i y_i$$

```
z = 0;  
for (i=0; i<n; i++)  
{  
    t = x[i]*y[i];  
    z = z+t;  
}
```

- Πόσος χρόνος απαιτείται ανά εντολή;
- Ποια είναι η συχνότητα εκτέλεσης της κάθε εντολής;

Ασυμπτωτική ανάλυση

Είναι η μελέτη του ρυθμού αύξησης του χρόνου εκτέλεσης (ή άλλης παραμέτρου απόδοσης) ως συνάρτηση του μεγέθους της εισόδου

Παράδειγμα: Υπολογισμός εσωτερικού γινομένου

$$z = x^T y = \sum_{i=1}^{i=n} x_i y_i$$

```
z = 0;  
for (i=0; i<n; i++)  
{  
    t = x[i]*y[i];  
    z = z+t;  
}
```

Χρόνος εκτέλεσης: $T(n) = c_1 + c_2n + c_3n + c_4n$

Ασυμπτωτική ανάλυση

Είναι η μελέτη του ρυθμού αύξησης του χρόνου εκτέλεσης (ή άλλης παραμέτρου απόδοσης) ως συνάρτηση του μεγέθους της εισόδου

Παράδειγμα: Υπολογισμός εσωτερικού γινομένου

$$z = x^T y = \sum_{i=1}^{i=n} x_i y_i$$

→ `z = 0;`
`for (i=0; i<n; i++)`
`{`
`t = x[i]*y[i];`
`z = z+t;`
`}`

Χρόνος εκτέλεσης: $T(n) = c_1 + c_2n + c_3n + c_4n$
↑

Ασυμπτωτική ανάλυση

Είναι η μελέτη του ρυθμού αύξησης του χρόνου εκτέλεσης (ή άλλης παραμέτρου απόδοσης) ως συνάρτηση του μεγέθους της εισόδου

Παράδειγμα: Υπολογισμός εσωτερικού γινομένου

$$z = x^T y = \sum_{i=1}^{i=n} x_i y_i$$

```
    z = 0;  
    → for (i=0; i<n; i++)  
      {  
          t = x[i]*y[i];  
          z = z+t;  
      }
```

Χρόνος εκτέλεσης: $T(n) = c_1 + c_2n + c_3n + c_4n$
↑

Ασυμπτωτική ανάλυση

Είναι η μελέτη του ρυθμού αύξησης του χρόνου εκτέλεσης (ή άλλης παραμέτρου απόδοσης) ως συνάρτηση του μεγέθους της εισόδου

Παράδειγμα: Υπολογισμός εσωτερικού γινομένου

$$z = x^T y = \sum_{i=1}^{i=n} x_i y_i$$

```
z = 0;  
for (i=0; i<n; i++)  
{  
     t = x[i]*y[i];  
    z = z+t;  
}
```

Χρόνος εκτέλεσης: $T(n) = c_1 + c_2n + c_3n + c_4n$



Ασυμπτωτική ανάλυση

Είναι η μελέτη του ρυθμού αύξησης του χρόνου εκτέλεσης (ή άλλης παραμέτρου απόδοσης) ως συνάρτηση του μεγέθους της εισόδου

Παράδειγμα: Υπολογισμός εσωτερικού γινομένου

$$z = x^T y = \sum_{i=1}^{i=n} x_i y_i$$

```
z = 0;  
for (i=0; i<n; i++)  
{  
    t = x[i]*y[i];  
    → z = z+t;  
}
```

Χρόνος εκτέλεσης: $T(n) = c_1 + c_2n + c_3n + c_4n$
↑

Ασυμπτωτική ανάλυση

Είναι η μελέτη του ρυθμού αύξησης του χρόνου εκτέλεσης (ή άλλης παραμέτρου απόδοσης) ως συνάρτηση του μεγέθους της εισόδου

Παράδειγμα: Υπολογισμός εσωτερικού γινομένου

$$z = x^T y = \sum_{i=1}^{i=n} x_i y_i$$

```
z = 0;  
for (i=0; i<n; i++)  
{  
    t = x[i]*y[i];  
    z = z+t;  
}
```

Χρόνος εκτέλεσης: $T(n) = c_1 + c_2n + c_3n + c_4n$

ασυμπτωτικά γραμμική

Ασυμπτωτική ανάλυση

Είναι η μελέτη του ρυθμού αύξησης του χρόνου εκτέλεσης (ή άλλης παραμέτρου απόδοσης) ως συνάρτηση του μεγέθους της εισόδου

Παράδειγμα: Υπολογισμός εσωτερικού γινομένου

Θέλουμε να εξαλείψουμε τις επουσιώδεις λεπτομέρειες

$$z = x^T y = \sum_{i=1}^{i=n} x_i y_i$$

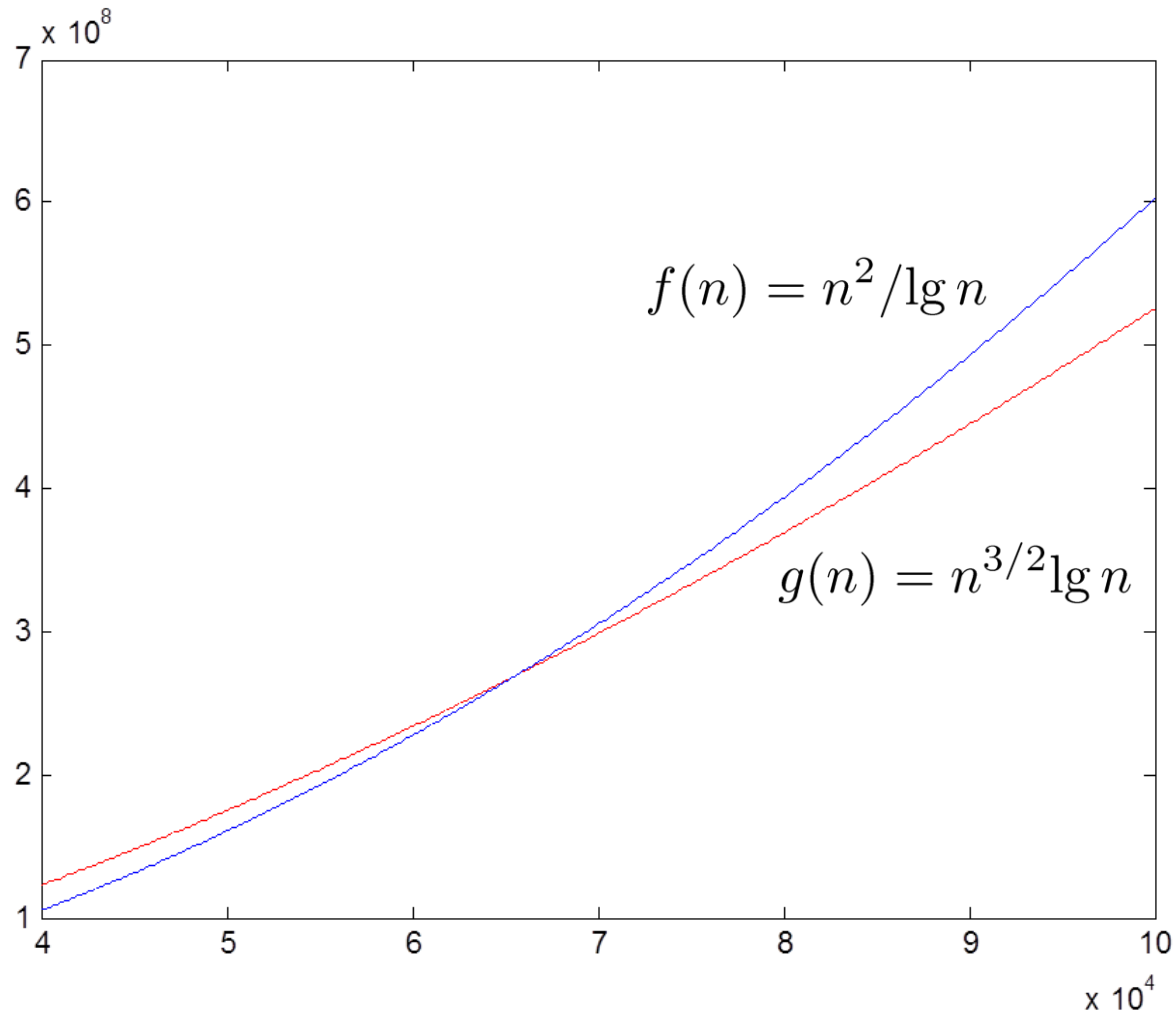
```
z = 0;  
for (i=0; i<n; i++)  
{  
    t = x[i]*y[i];  
    z = z+t;  
}
```

Χρόνος εκτέλεσης: $T(n) = c_1 + c_2n + c_3n + c_4n = \Theta(n)$

ασυμπτωτικά γραμμική

Ασυμπτωτική ανάλυση

Εξετάζουμε τη συμπεριφορά των συναρτήσεων όσο αυξάνει το μέγεθος της εισόδου

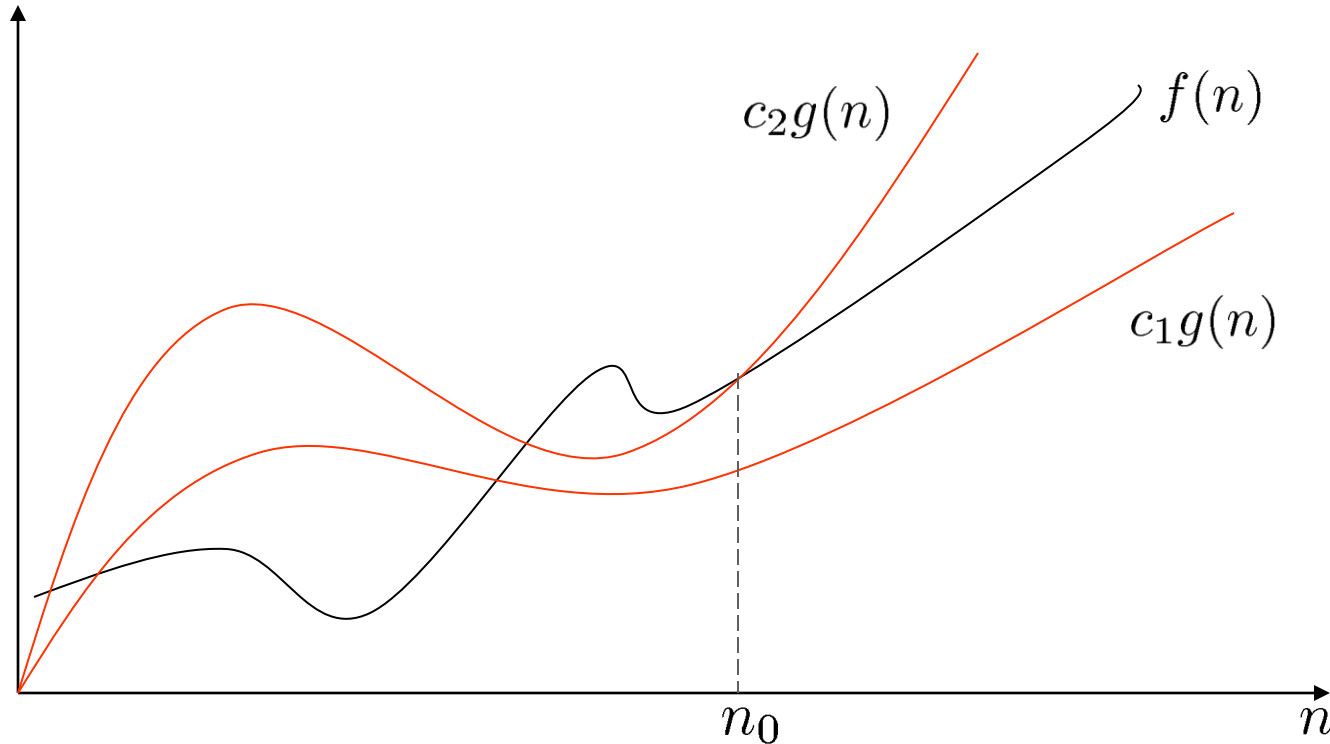


Ασυμπτωτική ανάλυση

Συμβολισμός Θ **ασυμπτωτικά αυστηρό φράγμα**

$f(n) = \Theta(g(n))$: Υπάρχουν θετικές σταθερές c_1, c_2 και n_0 τέτοιες ώστε

$$0 \leq c_1 g(n) \leq f(n) \leq c_2 g(n) \quad \text{για κάθε } n \geq n_0$$



Ασυμπτωτική ανάλυση

Συμβολισμός Θ **ασυμπτωτικά αυστηρό φράγμα**

$f(n) = \Theta(g(n))$: Υπάρχουν θετικές σταθερές c_1, c_2 και n_0 τέτοιες ώστε

$$0 \leq c_1 g(n) \leq f(n) \leq c_2 g(n) \quad \text{για κάθε } n \geq n_0$$

Παράδειγμα 1: $\frac{1}{2}n^2 - 3n = \Theta(n^2)$

Πρέπει να βρούμε θετικές σταθερές c_1, c_2 και n_0 , τέτοιες ώστε

$$c_1 n^2 \leq \frac{1}{2}n^2 - 3n \leq c_2 n^2 \quad \text{για κάθε } n \geq n_0.$$

Ασυμπτωτική ανάλυση

Συμβολισμός Θ ασυμπτωτικά αυστηρό φράγμα

$f(n) = \Theta(g(n))$: Υπάρχουν θετικές σταθερές c_1, c_2 και n_0 τέτοιες ώστε

$$0 \leq c_1 g(n) \leq f(n) \leq c_2 g(n) \quad \text{για κάθε } n \geq n_0$$

Παράδειγμα 1: $\frac{1}{2}n^2 - 3n = \Theta(n^2)$

Πρέπει να βρούμε θετικές σταθερές c_1, c_2 και n_0 , τέτοιες ώστε

(Σ) $c_1 n^2 \leq \frac{1}{2}n^2 - 3n \leq c_2 n^2 \quad \text{για κάθε } n \geq n_0.$

Ασυμπτωτική ανάλυση

Συμβολισμός Θ **ασυμπτωτικά αυστηρό φράγμα**

$f(n) = \Theta(g(n))$: Υπάρχουν θετικές σταθερές c_1, c_2 και n_0 τέτοιες ώστε

$$0 \leq c_1 g(n) \leq f(n) \leq c_2 g(n) \quad \text{για κάθε } n \geq n_0$$

Παράδειγμα 1: $\frac{1}{2}n^2 - 3n = \Theta(n^2)$

Πρέπει να βρούμε θετικές σταθερές c_1, c_2 και n_0 , τέτοιες ώστε

(Σ) $c_1 n^2 \leq \frac{1}{2}n^2 - 3n \leq c_2 n^2$ για κάθε $n \geq n_0$. Θέλουμε $c_2 \geq 1/2 - 3/n$

που ισχύει όταν $c_2 \geq 1/2$ και $n \geq 1$.

Επίσης θέλουμε $0 < c_1 \leq 1/2 - 3/n$. Έχουμε $1/2 - 3/n > 0 \Rightarrow n > 6$.

Επιλέγοντας $n = 7$ έχουμε $1/2 - 3/n = 1/2 - 3/7 = 1/14$.

Άρα για $n_0 = 7, c_1 = 1/14$ και $c_2 = 1/2$ ικανοποιείται η συνθήκη (Σ).

Ασυμπτωτική ανάλυση

Συμβολισμός Θ **ασυμπτωτικά αυστηρό φράγμα**

$f(n) = \Theta(g(n))$: Υπάρχουν θετικές σταθερές c_1, c_2 και n_0 τέτοιες ώστε

$$0 \leq c_1 g(n) \leq f(n) \leq c_2 g(n) \quad \text{για κάθε } n \geq n_0$$

Παράδειγμα 2: $6n^3 \neq \Theta(n^2)$

Έστω ότι $6n^3 = \Theta(n^2)$.

Τότε υπάρχουν θετικές σταθερές c_2 και n_0 με $6n^3 \leq c_2 n^2$ για κάθε $n \geq n_0$.

Δηλαδή $c_2 \geq 6n$ για κάθε $n \geq n_0$.

Αυτό είναι αδύνατο για σταθερό c_2 .

Ασυμπτωτική ανάλυση

Παράδειγμα: Η παρακάτω συνάρτηση εξετάζει τον πίνακα a με N ακέραιους και βρίσκει πόσες τριάδες δίνουν άθροισμα μηδέν

```
static public int countTriples(int[] a, int N){
    int count=0;
    for (int i=0; i<N; i++)
        for (int j=i+1; j<N; j++)
            for (int k=j+1; k<N; k++)
                if ( a[i]+a[j]+a[k] == 0) count++;
    return count;
}
```

Δοκιμάζει όλες τις τριάδες i, j, k με $0 \leq i < j < k \leq N - 1$

Ασυμπτωτική ανάλυση

Παράδειγμα: Η παρακάτω συνάρτηση εξετάζει τον πίνακα a με N ακέραιους και βρίσκει πόσες τριάδες δίνουν άθροισμα μηδέν

```
static public int countTriples(int[] a, int N){  
    int count=0;  
    for (int i=0; i<N; i++)  
        for (int j=i+1; j<N; j++)  
            for (int k=j+1; k<N; k++)  
                if ( a[i]+a[j]+a[k] == 0) count++;  
    return count;  
}
```

Δοκιμάζει όλες τις τριάδες i, j, k με $0 \leq i < j < k \leq N - 1$

$$\Rightarrow \binom{N}{3} = \frac{N(N-1)(N-2)}{6} \text{ τριάδες}$$

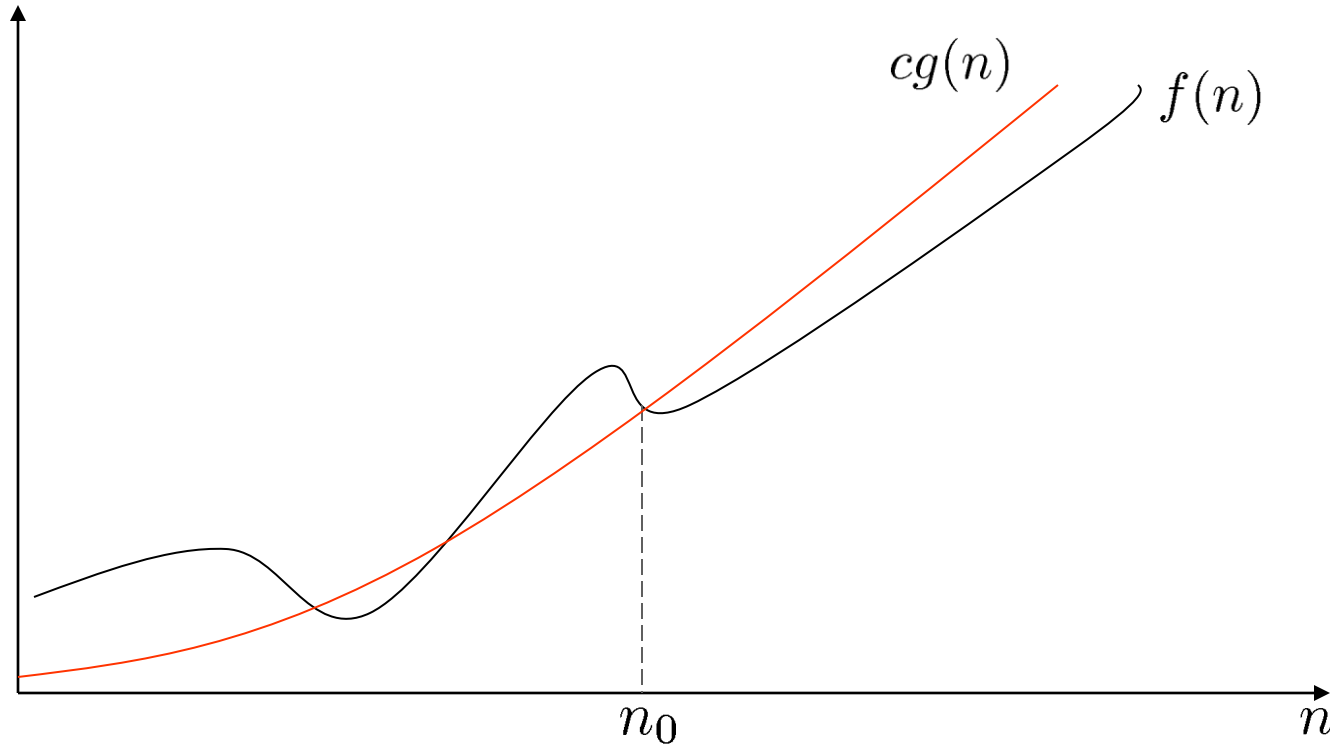
$$\Rightarrow \Theta(N^3) \text{ χρόνος εκτέλεσης}$$

Ασυμπτωτική ανάλυση

Συμβολισμός O **ασυμπτωτικό άνω φράγμα**

$f(n) = O(g(n))$: Υπάρχουν θετικές σταθερές c και n_0 τέτοιες ώστε

$$0 \leq f(n) \leq cg(n) \quad \text{για κάθε } n \geq n_0$$

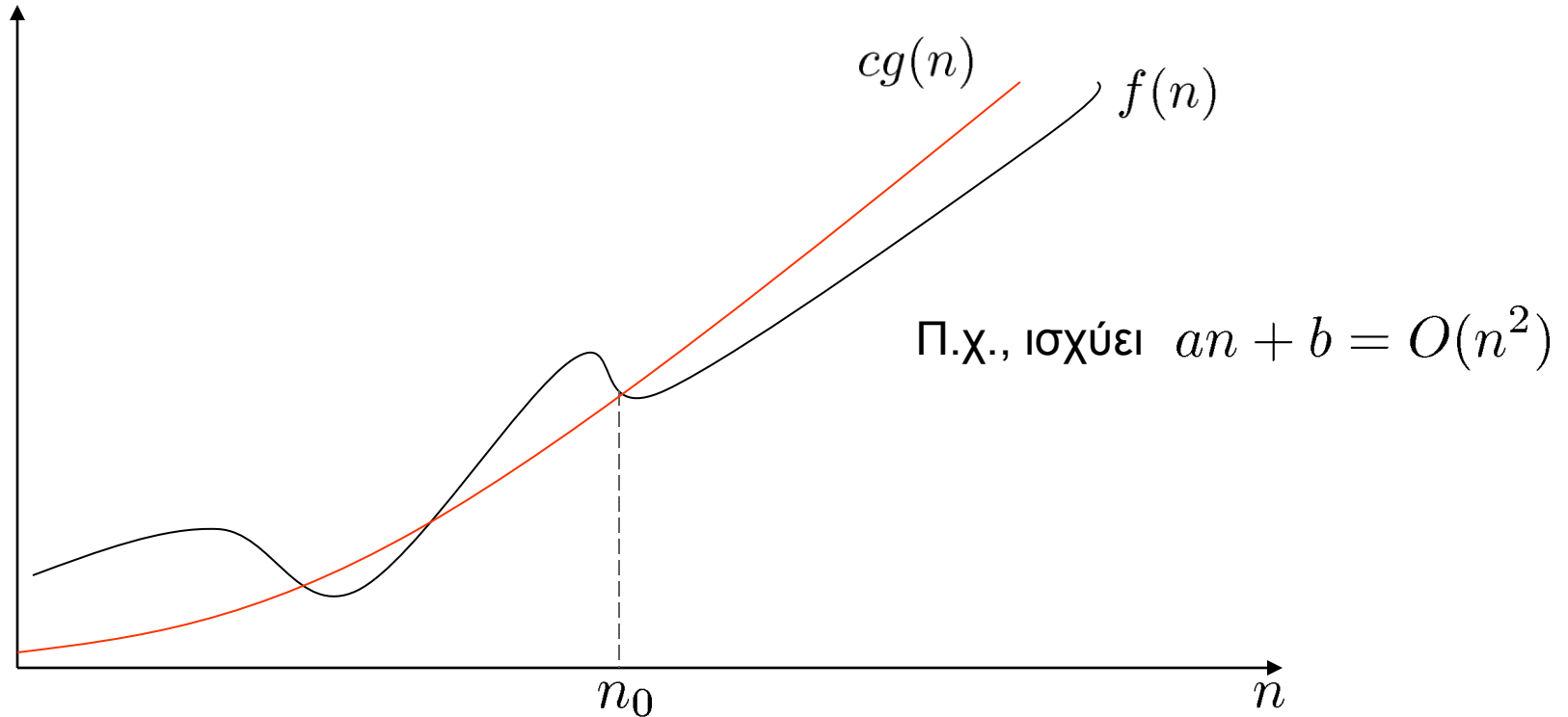


Ασυμπτωτική ανάλυση

Συμβολισμός O **ασυμπτωτικό άνω φράγμα**

$f(n) = O(g(n))$: Υπάρχουν θετικές σταθερές c και n_0 τέτοιες ώστε

$$0 \leq f(n) \leq cg(n) \quad \text{για κάθε } n \geq n_0$$

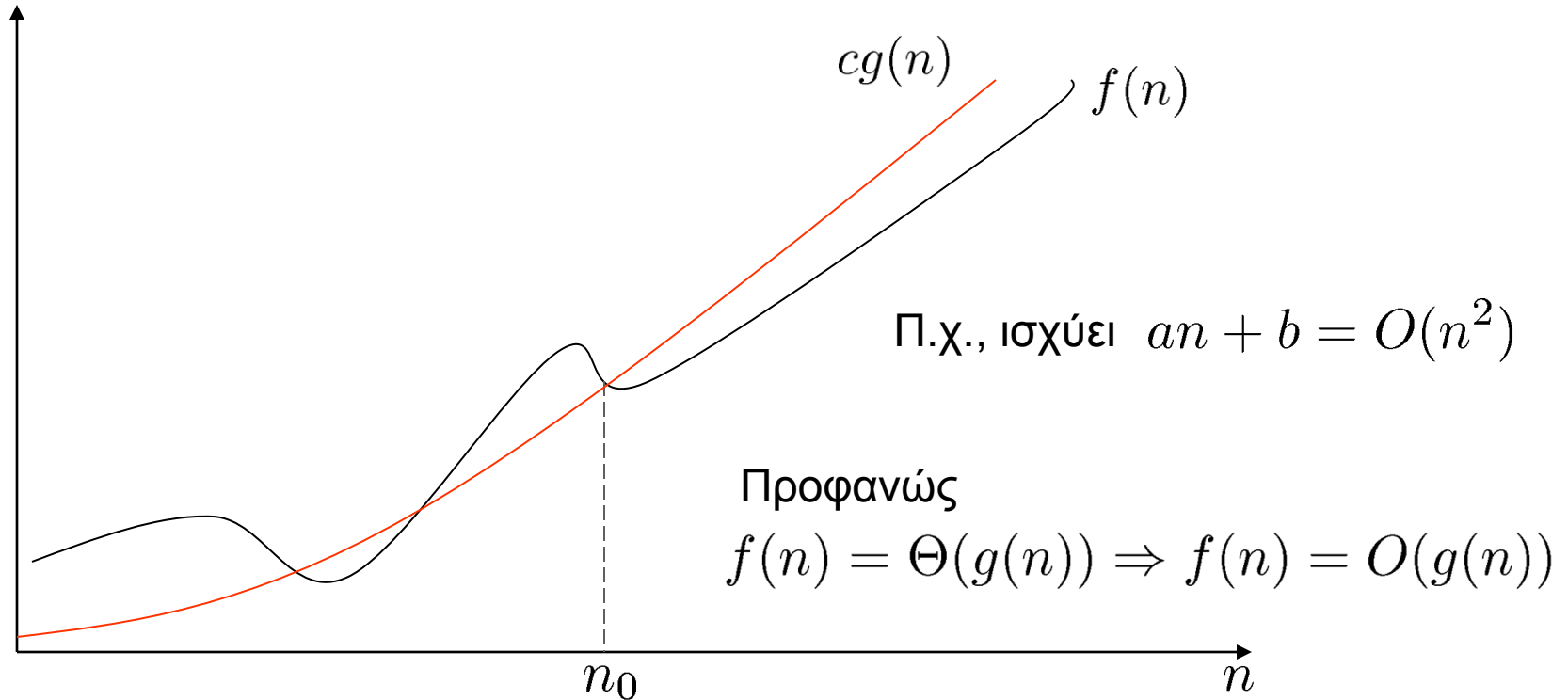


Ασυμπτωτική ανάλυση

Συμβολισμός O **ασυμπτωτικό άνω φράγμα**

$f(n) = O(g(n))$: Υπάρχουν θετικές σταθερές c και n_0 τέτοιες ώστε

$$0 \leq f(n) \leq cg(n) \quad \text{για κάθε } n \geq n_0$$



Ασυμπτωτική ανάλυση

Συμβολισμός O ασυμπτωτικό άνω φράγμα

$f(n) = O(g(n))$: Υπάρχουν θετικές σταθερές c και n_0 τέτοιες ώστε

$$0 \leq f(n) \leq cg(n) \quad \text{για κάθε } n \geq n_0$$

Έστω ότι για τις συναρτήσεις $f(n)$ και $g(n)$ υπάρχει το όριο

$$\lim_{n \rightarrow \infty} \frac{f(n)}{g(n)} \quad \text{και έχει τιμή } \leq c \quad \text{για κάποια σταθερά } c$$

Τότε $f(n) = O(g(n))$

Ασυμπτωτική ανάλυση

Συμβολισμός O **ασυμπτωτικό άνω φράγμα**

$f(n) = O(g(n))$: Υπάρχουν θετικές σταθερές c και n_0 τέτοιες ώστε

$$0 \leq f(n) \leq cg(n) \quad \text{για κάθε } n \geq n_0$$

Έστω ότι για τις συναρτήσεις $f(n)$ και $g(n)$ υπάρχει το όριο

$$\lim_{n \rightarrow \infty} \frac{f(n)}{g(n)} \quad \text{και έχει τιμή } \leq c \quad \text{για κάποια σταθερά } c$$

Τότε $f(n) = O(g(n))$

Παράδειγμα $2n^3 + 5n^2 \log n = O(n^3)$

$$\text{επειδή} \quad \lim_{n \rightarrow \infty} \frac{2n^3 + 5n^2 \log n}{n^3} = 2$$

Ασυμπτωτική ανάλυση

Συμβολισμός O **ασυμπτωτικό άνω φράγμα**

$f(n) = O(g(n))$: Υπάρχουν θετικές σταθερές c και n_0 τέτοιες ώστε

$$0 \leq f(n) \leq cg(n) \quad \text{για κάθε } n \geq n_0$$

Έστω ότι για τις συναρτήσεις $f(n)$ και $g(n)$ υπάρχει το όριο

$$\lim_{n \rightarrow \infty} \frac{f(n)}{g(n)} \quad \text{και έχει τιμή } \leq c \quad \text{για κάποια σταθερά } c$$

Τότε $f(n) = O(g(n))$

Παράδειγμα $\frac{n}{2 \log n} = O(n)$

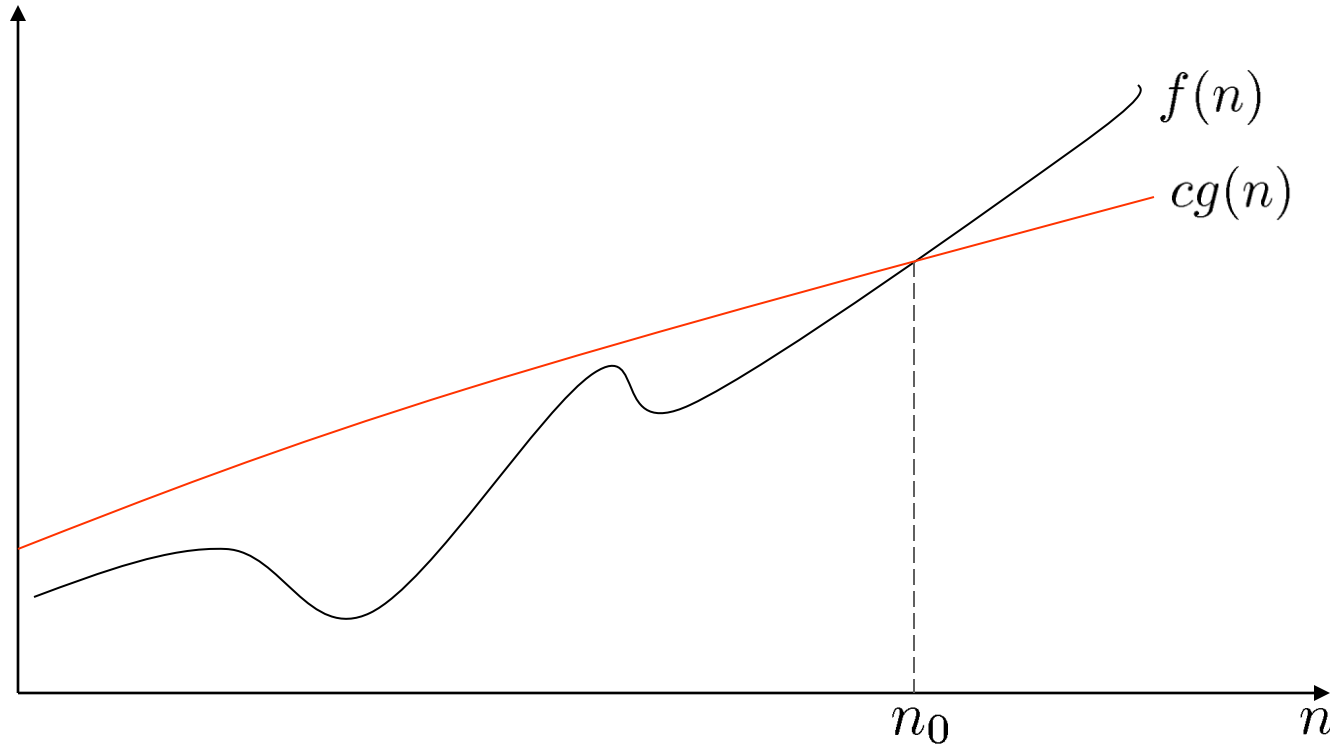
$$\text{επειδή } \lim_{n \rightarrow \infty} \frac{\frac{n}{2 \log n}}{n} = 0$$

Ασυμπτωτική ανάλυση

Συμβολισμός Ω ασυμπτωτικό κάτω φράγμα

$f(n) = \Omega(g(n))$: Υπάρχουν θετικές σταθερές c και n_0 τέτοιες ώστε

$$0 \leq cg(n) \leq f(n) \quad \text{για κάθε } n \geq n_0$$



Ασυμπτωτική ανάλυση

Συμβολισμός Ω ασυμπτωτικό κάτω φράγμα

$f(n) = \Omega(g(n))$: Υπάρχουν θετικές σταθερές c και n_0 τέτοιες ώστε

$$0 \leq cg(n) \leq f(n) \quad \text{για κάθε } n \geq n_0$$

Έστω ότι για τις συναρτήσεις $f(n)$ και $g(n)$ υπάρχει το όριο

$$\lim_{n \rightarrow \infty} \frac{g(n)}{f(n)} \quad \text{και έχει τιμή } \leq c \quad \text{για κάποια σταθερά } c$$

Τότε $f(n) = \Omega(g(n))$

Ισχύει $f(n) = \Omega(g(n)) \Leftrightarrow g(n) = O(f(n))$

Ασυμπτωτική ανάλυση

Συμβολισμός Ω ασυμπτωτικό κάτω φράγμα

$f(n) = \Omega(g(n))$: Υπάρχουν θετικές σταθερές c και n_0 τέτοιες ώστε

$$0 \leq cg(n) \leq f(n) \quad \text{για κάθε } n \geq n_0$$

Έστω ότι για τις συναρτήσεις $f(n)$ και $g(n)$ υπάρχει το όριο

$$\lim_{n \rightarrow \infty} \frac{g(n)}{f(n)} \quad \text{και έχει τιμή} \leq c \quad \text{για κάποια σταθερά } c$$

Τότε $f(n) = \Omega(g(n))$

Παράδειγμα $2n^3 + 5n^2 \log n = \Omega(n^3)$

$$\text{επειδή} \quad \lim_{n \rightarrow \infty} \frac{n^3}{2n^3 + 5n^2 \log n} = \frac{1}{2}$$

Ασυμπτωτική ανάλυση

Συμβολισμός Ω ασυμπτωτικό κάτω φράγμα

$f(n) = \Omega(g(n))$: Υπάρχουν θετικές σταθερές c και n_0 τέτοιες ώστε

$$0 \leq cg(n) \leq f(n) \quad \text{για κάθε } n \geq n_0$$

Έστω ότι για τις συναρτήσεις $f(n)$ και $g(n)$ υπάρχει το όριο

$$\lim_{n \rightarrow \infty} \frac{g(n)}{f(n)} \quad \text{και έχει τιμή } \leq c \quad \text{για κάποια σταθερά } c$$

Τότε $f(n) = \Omega(g(n))$

Παράδειγμα $2n \log n = \Omega(n)$

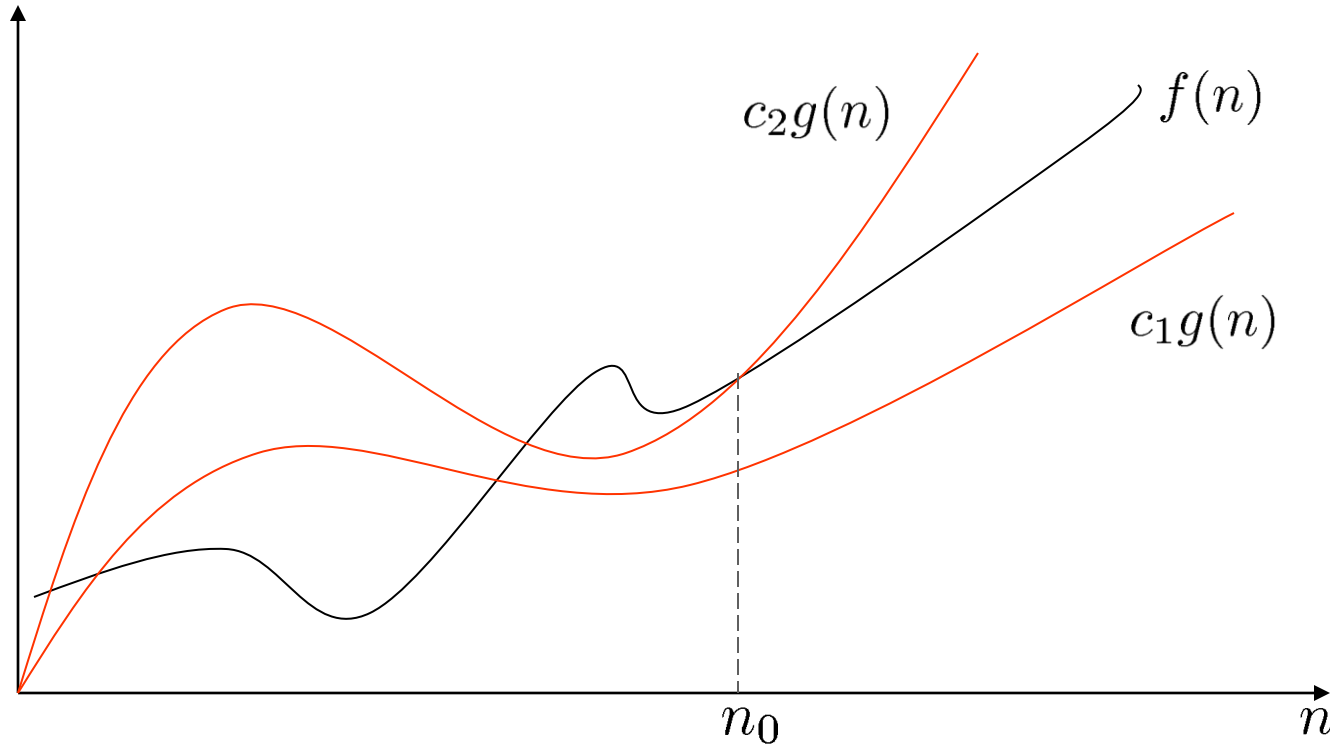
$$\text{επειδή } \lim_{n \rightarrow \infty} \frac{n}{2n \log n} = 0$$

Ασυμπτωτική ανάλυση

Συμβολισμός Θ **ασυμπτωτικά αυστηρό φράγμα**

$f(n) = \Theta(g(n))$: Υπάρχουν θετικές σταθερές c_1, c_2 και n_0 τέτοιες ώστε

$$0 \leq c_1 g(n) \leq f(n) \leq c_2 g(n) \quad \text{για κάθε } n \geq n_0$$



Ασυμπτωτική ανάλυση

Συμβολισμός Θ **ασυμπτωτικά αυστηρό φράγμα**

$f(n) = \Theta(g(n))$: Υπάρχουν θετικές σταθερές c_1, c_2 και n_0 τέτοιες ώστε

$$0 \leq c_1 g(n) \leq f(n) \leq c_2 g(n) \text{ για κάθε } n \geq n_0$$

Ισχύει $f(n) = \Theta(g(n))$ αν και μόνο αν $f(n) = O(g(n))$ και $f(n) = \Omega(g(n))$

Έστω ότι για τις συναρτήσεις $f(n)$ και $g(n)$ υπάρχουν τα όρια

$$\lim_{n \rightarrow \infty} \frac{f(n)}{g(n)} \neq 0, \quad \lim_{n \rightarrow \infty} \frac{g(n)}{f(n)} \neq \infty \quad \text{όπου} \quad \lim_{n \rightarrow \infty} \frac{f(n)}{g(n)} \leq c \quad \text{και} \quad \lim_{n \rightarrow \infty} \frac{g(n)}{f(n)} \leq c$$

για κάποια σταθερά c . Τότε $f(n) = \Theta(g(n))$

Ασυμπτωτική ανάλυση

Συμβολισμός o **ασυμπτωτικά μη αυστηρό άνω φράγμα**

$f(n) = o(g(n))$: Για οποιαδήποτε σταθερά $c > 0$, υπάρχει σταθερά $n_0 > 0$

τέτοια ώστε $0 \leq f(n) < cg(n)$ για κάθε $n \geq n_0$

Παράδειγμα: $2n = o(n^2)$ αλλά $2n^2 \neq o(n^2)$

Έστω ότι για τις συναρτήσεις $f(n)$ και $g(n)$ υπάρχει το όριο

$\lim_{n \rightarrow \infty} \frac{f(n)}{g(n)}$ όπου $\lim_{n \rightarrow \infty} \frac{f(n)}{g(n)} = 0$. Τότε $f(n) = o(g(n))$

Ασυμπτωτική ανάλυση

Συμβολισμός ω **ασυμπτωτικά μη αυστηρό κάτω φράγμα**

$f(n) = \omega(g(n))$: Για οποιαδήποτε σταθερά $c > 0$, υπάρχει σταθερά $n_0 > 0$

τέτοια ώστε $0 \leq cg(n) < f(n)$ για κάθε $n \geq n_0$

Παράδειγμα: $n^2/2 = \omega(n)$ αλλά $n^2/2 \neq \omega(n^2)$

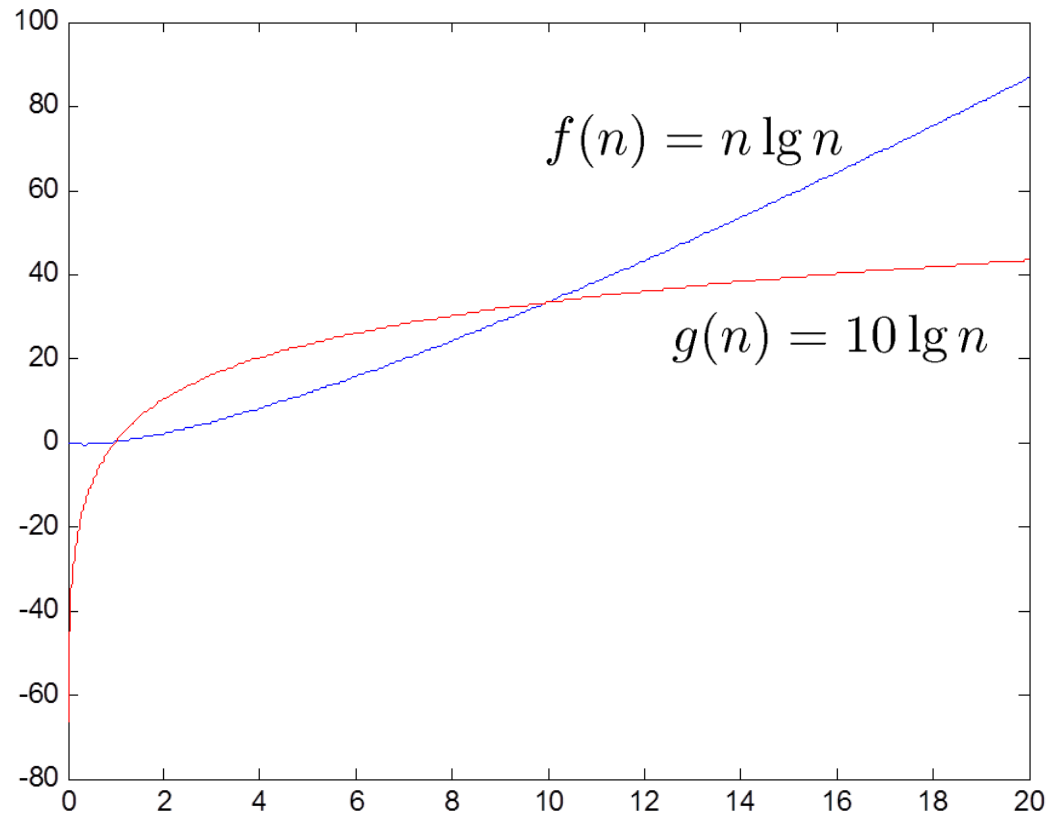
Έστω ότι για τις συναρτήσεις $f(n)$ και $g(n)$ υπάρχει το όριο

$\lim_{n \rightarrow \infty} \frac{g(n)}{f(n)}$ όπου $\lim_{n \rightarrow \infty} \frac{g(n)}{f(n)} = 0$. Τότε $f(n) = \omega(g(n))$

Ισχύει $f(n) = \omega(g(n)) \Leftrightarrow g(n) = o(f(n))$

Ασυμπτωτική ανάλυση

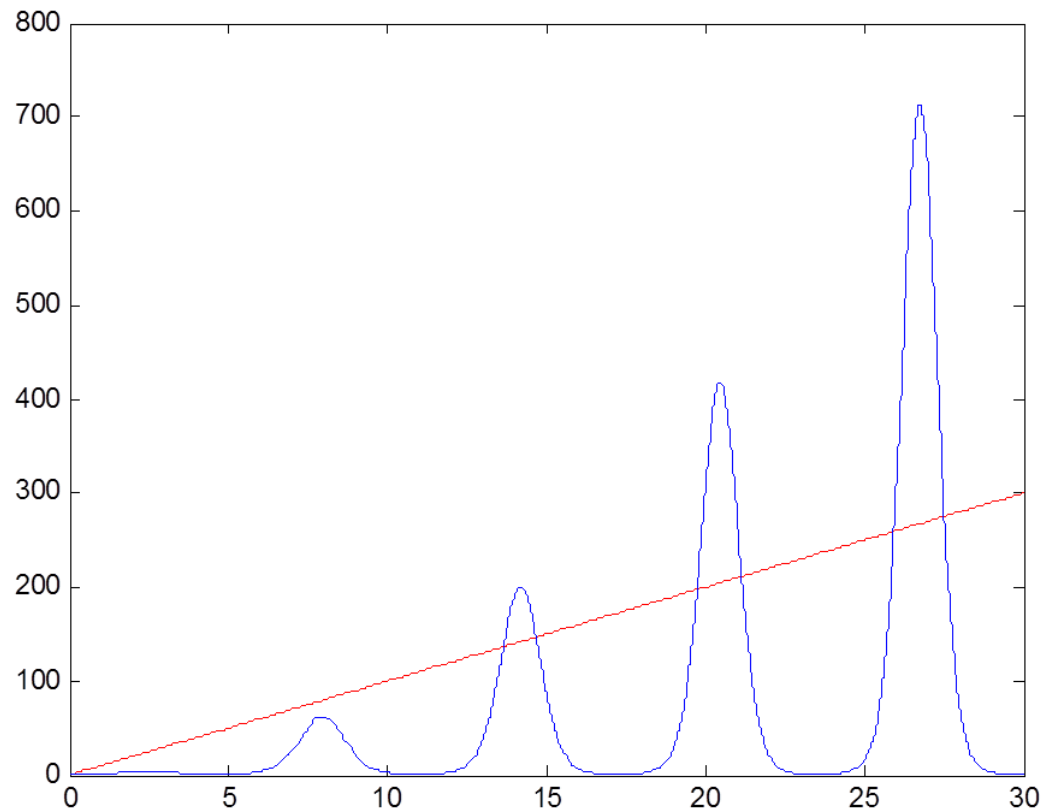
Προσοχή! Ακόμα και δύο γνησίως αύξουσες συναρτήσεις μπορεί να έχουν πολλά σημεία τομής



Ασυμπτωτική ανάλυση

Προσοχή! Δεν είναι όλες οι συναρτήσεις ασυμπτωτικά συγκρίσιμες

π.χ., $f(n) = n^{1+\sin n}$ και $g(n) = n$



Ασυμπτωτική ανάλυση

Ιδιότητες

Μεταβατικότητα:

$$f(n) = \Theta(g(n)) \text{ και } g(n) = \Theta(h(n)) \Rightarrow f(n) = \Theta(h(n))$$

Αυτοπάθεια:

$$f(n) = \Theta(f(n))$$

Συμμετρία:

$$f(n) = \Theta(g(n)) \text{ όταν και μόνο όταν } g(n) = \Theta(f(n))$$

Ασυμπτωτική ανάλυση

Ιδιότητες

Μεταβατικότητα:

$$f(n) = \Theta(g(n)) \text{ και } g(n) = \Theta(h(n)) \Rightarrow f(n) = \Theta(h(n))$$

Αυτοπάθεια:

$$f(n) = \Theta(f(n))$$

Συμμετρία:

$$f(n) = \Theta(g(n)) \text{ όταν και μόνο όταν } g(n) = \Theta(f(n))$$

Μεταβατικότητα και αυτοπάθεια ισχύουν και για τους συμβολισμούς O , Ω , o και ω .
Στη θέση της συμμετρίας έχουμε:

$$f(n) = O(g(n)) \text{ όταν και μόνο όταν } g(n) = \Omega(f(n))$$

Ασυμπτωτική ανάλυση

Παραδείγματα

Θέλουμε να διατάξουμε τις παρακάτω συναρτήσεις ως προς το ρυθμό αύξησης, από τη μικρότερη ως τη μεγαλύτερη

$$f_1(n) = n^{5/4}, f_2(n) = n \lg n, f_3(n) = n!$$

$$f_4(n) = 10^n, f_5 = n^{\lg n}$$

Ασυμπτωτική ανάλυση

Παραδείγματα

Θέλουμε να διατάξουμε τις παρακάτω συναρτήσεις ως προς το ρυθμό αύξησης, από τη μικρότερη ως τη μεγαλύτερη

$$f_1(n) = n^{5/4}, f_2(n) = n \lg n, f_3(n) = n!$$

$$f_4(n) = 10^n, f_5 = n^{\lg n}$$

Έχουμε $f_2(n) = o(f_1(n))$: Για $n = 2^{20}$

$$f_1(n) = (2^{20})^{5/4} = 2^{25} = 2^5 \cdot 2^{20} \text{ και } f_2(n) = 20 \cdot 2^{20}$$

$$\text{Γενικά } \lim_{n \rightarrow \infty} \frac{n \lg n}{n^{5/4}} = \lim_{n \rightarrow \infty} \frac{\lg n}{n^{1/4}} = \lim_{n \rightarrow \infty} \frac{(\lg n)'}{(n^{1/4})'} = \lim_{n \rightarrow \infty} \frac{4}{n^{1/4}} = 0$$

Διάταξη f_2, f_1

Ασυμπτωτική ανάλυση

Παραδείγματα

Θέλουμε να διατάξουμε τις παρακάτω συναρτήσεις ως προς το ρυθμό αύξησης, από τη μικρότερη ως τη μεγαλύτερη

$$f_1(n) = n^{5/4}, f_2(n) = n \lg n, f_3(n) = n!$$

$$f_4(n) = 10^n, f_5 = n^{\lg n}$$

Έχουμε $f_1(n) = o(f_4(n))$:

$$\text{Για } n \geq 1, \lg f_1(n) = \frac{5}{4} \lg n < \lg f_4(n) = \lg 10 \cdot n$$

$$\text{Έχουμε } \lim_{n \rightarrow \infty} \frac{n^{5/4}}{10^n} = 0$$

Διάταξη f_2, f_1, f_4

Ασυμπτωτική ανάλυση

Παραδείγματα

Θέλουμε να διατάξουμε τις παρακάτω συναρτήσεις ως προς το ρυθμό αύξησης, από τη μικρότερη ως τη μεγαλύτερη

$$f_1(n) = n^{5/4}, f_2(n) = n \lg n, f_3(n) = n!$$

$$f_4(n) = 10^n, f_5 = n^{\lg n}$$

Έχουμε $f_5(n) = o(f_4(n))$:

$$\text{Για } n \geq 1, \lg f_5(n) = (\lg n)^2 < \lg f_4(n) = \lg 10 \cdot n$$

$$\text{Έχουμε } \lim_{n \rightarrow \infty} \frac{n^{\lg n}}{10^n} = 0$$

$$\text{Διάταξη } f_2, f_1, f_4, f_5 = o(f_4)$$

Ασυμπτωτική ανάλυση

Παραδείγματα

Θέλουμε να διατάξουμε τις παρακάτω συναρτήσεις ως προς το ρυθμό αύξησης, από τη μικρότερη ως τη μεγαλύτερη

$$f_1(n) = n^{5/4}, f_2(n) = n \lg n, f_3(n) = n!$$

$$f_4(n) = 10^n, f_5 = n^{\lg n}$$

Έχουμε $f_4(n) = o(f_3(n))$:

$$\text{Για } n \geq 1, \lg f_4(n) = \lg 10 \cdot n < 4 \lg f_3(n) = \Theta(n \lg n)$$

$$\text{Έχουμε } \lim_{n \rightarrow \infty} \frac{10^n}{n!} = 0$$

$$\text{Διάταξη } f_2, f_1, f_4, f_3, f_5 = o(f_4)$$

Ασυμπτωτική ανάλυση

Παραδείγματα

Θέλουμε να διατάξουμε τις παρακάτω συναρτήσεις ως προς το ρυθμό αύξησης, από τη μικρότερη ως τη μεγαλύτερη

$$f_1(n) = n^{5/4}, f_2(n) = n \lg n, f_3(n) = n!$$

$$f_4(n) = 10^n, f_5 = n^{\lg n}$$

$$\text{Έχουμε } f_1(n) = o(f_5(n))$$

$$\text{Για } n \geq 4, \quad \lg f_1(n) = \frac{5}{4} \lg n < \lg f_5(n) = (\lg n)^2$$

$$\text{Έχουμε } \lim_{n \rightarrow \infty} \frac{n^{5/4}}{n^{\lg n}} = 0$$

$$\text{Διάταξη } f_2, f_1, f_4, f_3, \quad f_1 = o(f_5), f_5 = o(f_4)$$

Ασυμπτωτική ανάλυση

Παραδείγματα

Θέλουμε να διατάξουμε τις παρακάτω συναρτήσεις ως προς το ρυθμό αύξησης, από τη μικρότερη ως τη μεγαλύτερη

$$f_1(n) = n^{5/4}, f_2(n) = n \lg n, f_3(n) = n!$$

$$f_4(n) = 10^n, f_5 = n^{\lg n}$$

Επομένως έχουμε

$$f_2, f_1, f_5, f_4, f_3$$

Ασυμπτωτική ανάλυση

Πολυωνυμικός Αλγόριθμος

Έχει πολυωνυμικό χρόνο εκτέλεσης $T(n)$, δηλαδή

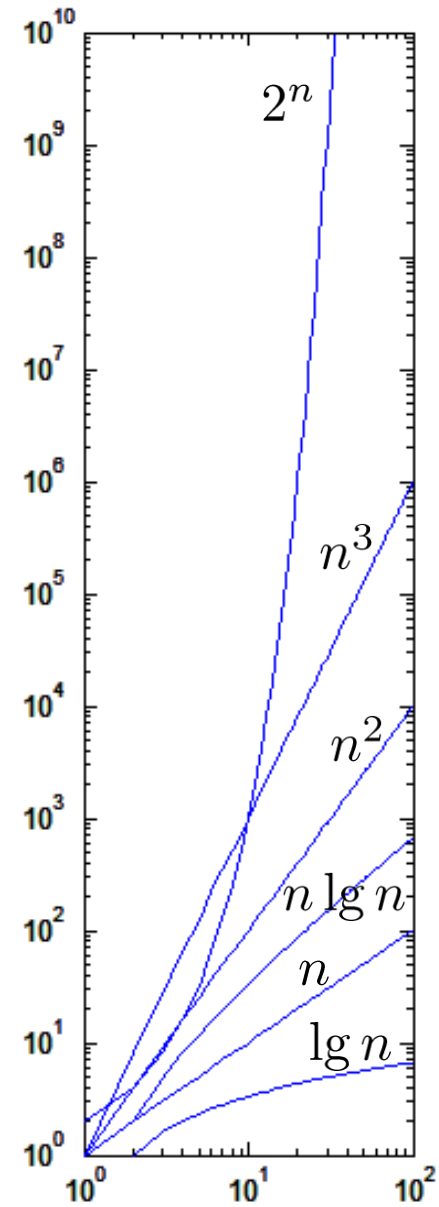
$$T(n) = O(n^k) \quad \text{όπου } k \text{ είναι μία θετική σταθερά}$$

Κλάση πολυπλοκότητας **P**: περιλαμβάνει τα προβλήματα που επιδέχονται λύση σε **πολυωνυμικό χρόνο** (δηλ. υπάρχουν πολυωνυμικοί αλγόριθμοι που τα επιλύουν)
(**P** = polynomial)

Ασυμπτωτική ανάλυση

Πρακτικές πολυπλοκότητες

$\lg n$	n	$n \lg n$	n^2	n^3	2^n	$n!$
0	1	0	1	1	2	1
1	2	2	4	8	4	2
2	4	8	16	64	16	24
3	8	24	64	512	256	40320
4	16	64	256	4096	65536	20922789888000



λογαριθμική κλίμακα

Ασυμπτωτική ανάλυση

Χρήσιμες συναρτήσεις στην ανάλυση αλγορίθμων

$\lfloor x \rfloor$ κάτω ακέραιο μέρος : μεγαλύτερος ακέραιος $\leq x$

$\lceil x \rceil$ άνω ακέραιο μέρος : μικρότερος ακέραιος $\geq x$

$\ln N$ φυσικός λογάριθμος : x τέτοιο ώστε $e^x = N$

$\lg N$ δυαδικός λογάριθμος : x τέτοιο ώστε $2^x = N$

$\lfloor \lg N \rfloor$ ακέραιος δυαδικός λογάριθμος : (αριθμός δυαδικών ψηφίων στη
δυαδική αναπαράσταση του N) - 1

$H_N = 1 + 1/2 + 1/3 + \dots + 1/N \approx \ln N$ N-οστός αρμονικός αριθμός

Ασυμπτωτική ανάλυση

Χρήσιμες συναρτήσεις στην ανάλυση αλγορίθμων

$$\binom{N}{k} = \frac{N!}{(N-k)!k!} \quad \text{διωνυμικοί συντελεστές}$$

$$\binom{N}{k} \approx \frac{N^k}{k!} \quad \text{για μικρή σταθερά } k$$

Προσέγγιση Stirling $\lg N! = \lg 1 + \lg 2 + \lg 3 + \dots + \lg N \approx N \lg N$

Γεωμετρικό άθροισμα $1 + 2 + 4 + 8 + \dots + 2^n = 2 \cdot 2^n - 1$

Παράδειγμα: Δυαδική Αναζήτηση

Μας δίνεται μία ακολουθία από n ακέραιους $A = a_1, a_2, \dots, a_n$
και ένας ακέραιος v

Θέλουμε να βρούμε ένα $a_j \in A$ τέτοιο ώστε $a_j = v$

Δυαδική αναζήτηση: προϋποθέτει να έχει γίνει ταξινόμηση της ακολουθίας

$$\text{δηλαδή } a_1 \leq a_2 \leq \dots \leq a_n$$

Κάθε βήμα είτε βρίσκει το v είτε αποκλείει τα μισά σχεδόν στοιχεία της ακολουθίας που απομένουν.

Παράδειγμα: Δυαδική Αναζήτηση

Μας δίνεται μία ακολουθία από n ακέραιους $A = a_1, a_2, \dots, a_n$
και ένας ακέραιος v

Θέλουμε να βρούμε ένα $a_j \in A$ τέτοιο ώστε $a_j = v$

Δυαδική αναζήτηση: προϋποθέτει να έχει γίνει ταξινόμηση της ακολουθίας

$$\text{δηλαδή } a_1 \leq a_2 \leq \dots \leq a_n$$

Κάθε βήμα είτε βρίσκει το v είτε αποκλείει τα μισά σχεδόν στοιχεία της ακολουθίας που απομένουν.

θέση	<u>1</u>	<u>2</u>	<u>3</u>	<u>4</u>	<u>5</u>	<u>6</u>	<u>7</u>	<u>8</u>	<u>9</u>	<u>10</u>	<u>11</u>	<u>12</u>	<u>13</u>	<u>14</u>	<u>15</u>
στοιχείο	3	8	9	12	28	33	45	56	88	90	102	128	233	245	356

Παράδειγμα: Δυαδική Αναζήτηση

Μας δίνεται μία ακολουθία από n ακέραιους $A = a_1, a_2, \dots, a_n$
και ένας ακέραιος v

Θέλουμε να βρούμε ένα $a_j \in A$ τέτοιο ώστε $a_j = v$

Δυαδική αναζήτηση: προϋποθέτει να έχει γίνει ταξινόμηση της ακολουθίας

$$\text{δηλαδή } a_1 \leq a_2 \leq \dots \leq a_n$$

Κάθε βήμα είτε βρίσκει το v είτε αποκλείει τα μισά σχεδόν στοιχεία της ακολουθίας που απομένουν.

θέση	<u>1</u>	<u>2</u>	<u>3</u>	<u>4</u>	<u>5</u>	<u>6</u>	<u>7</u>	<u>8</u>	<u>9</u>	<u>10</u>	<u>11</u>	<u>12</u>	<u>13</u>	<u>14</u>	<u>15</u>
στοιχείο	3	8	9	12	28	33	45	56	88	90	102	128	233	245	356

$v = 111$



$$\text{θέση μέσου στοιχείου} = (15+1)/2 = 8 \quad a_8 = 56 < v$$

Παράδειγμα: Δυαδική Αναζήτηση

Μας δίνεται μία ακολουθία από n ακέραιους $A = a_1, a_2, \dots, a_n$
και ένας ακέραιος v

Θέλουμε να βρούμε ένα $a_j \in A$ τέτοιο ώστε $a_j = v$

Δυαδική αναζήτηση: προϋποθέτει να έχει γίνει ταξινόμηση της ακολουθίας

$$\text{δηλαδή } a_1 \leq a_2 \leq \dots \leq a_n$$

Κάθε βήμα είτε βρίσκει το v είτε αποκλείει τα μισά σχεδόν στοιχεία της ακολουθίας που απομένουν.

θέση	<u>1</u>	<u>2</u>	<u>3</u>	<u>4</u>	<u>5</u>	<u>6</u>	<u>7</u>	<u>8</u>	<u>9</u>	<u>10</u>	<u>11</u>	<u>12</u>	<u>13</u>	<u>14</u>	<u>15</u>
στοιχείο	3	8	9	12	28	33	45	56	88	90	102	128	233	245	356

$v = 111$

$$\text{θέση μέσου στοιχείου} = (15+1)/2 = 8 \quad a_8 = 56 < v$$

Παράδειγμα: Δυαδική Αναζήτηση

Μας δίνεται μία ακολουθία από n ακέραιους $A = a_1, a_2, \dots, a_n$
και ένας ακέραιος v

Θέλουμε να βρούμε ένα $a_j \in A$ τέτοιο ώστε $a_j = v$

Δυαδική αναζήτηση: προϋποθέτει να έχει γίνει ταξινόμηση της ακολουθίας

$$\text{δηλαδή } a_1 \leq a_2 \leq \dots \leq a_n$$

Κάθε βήμα είτε βρίσκει το v είτε αποκλείει τα μισά σχεδόν στοιχεία της ακολουθίας που απομένουν.

θέση	<u>1</u>	<u>2</u>	<u>3</u>	<u>4</u>	<u>5</u>	<u>6</u>	<u>7</u>	<u>8</u>	<u>9</u>	<u>10</u>	<u>11</u>	<u>12</u>	<u>13</u>	<u>14</u>	<u>15</u>
στοιχείο	3	8	9	12	28	33	45	56	88	90	102	128	233	245	356

$v = 111$

$$\text{θέση μέσου στοιχείου} = (9+15)/2 = 12 \quad a_{12} = 128 > v$$

Παράδειγμα: Δυαδική Αναζήτηση

Μας δίνεται μία ακολουθία από n ακέραιους $A = a_1, a_2, \dots, a_n$
και ένας ακέραιος v

Θέλουμε να βρούμε ένα $a_j \in A$ τέτοιο ώστε $a_j = v$

Δυαδική αναζήτηση: προϋποθέτει να έχει γίνει ταξινόμηση της ακολουθίας

$$\text{δηλαδή } a_1 \leq a_2 \leq \dots \leq a_n$$

Κάθε βήμα είτε βρίσκει το v είτε αποκλείει τα μισά σχεδόν στοιχεία της ακολουθίας που απομένουν.

θέση	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
στοιχείο	3	8	9	12	28	33	45	56	88	90	102	128	233	245	356

$v = 111$

↑

$$\text{θέση μέσου στοιχείου} = (9+15)/2 = 12 \quad a_{12} = 128 > v$$

Παράδειγμα: Δυαδική Αναζήτηση

Μας δίνεται μία ακολουθία από n ακέραιους $A = a_1, a_2, \dots, a_n$
και ένας ακέραιος v

Θέλουμε να βρούμε ένα $a_j \in A$ τέτοιο ώστε $a_j = v$

Δυαδική αναζήτηση: προϋποθέτει να έχει γίνει ταξινόμηση της ακολουθίας

$$\text{δηλαδή } a_1 \leq a_2 \leq \dots \leq a_n$$

Κάθε βήμα είτε βρίσκει το v είτε αποκλείει τα μισά σχεδόν στοιχεία της ακολουθίας που απομένουν.

θέση	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
στοιχείο	3	8	9	12	28	33	45	56	88	90	102	128	233	245	356
$v = 111$										↑					

$$\text{θέση μέσου στοιχείου} = (9+11)/2 = 10 \quad a_{10} = 90 < v$$

Παράδειγμα: Δυαδική Αναζήτηση

Μας δίνεται μία ακολουθία από n ακέραιους $A = a_1, a_2, \dots, a_n$
και ένας ακέραιος v

Θέλουμε να βρούμε ένα $a_j \in A$ τέτοιο ώστε $a_j = v$

Δυαδική αναζήτηση: προϋποθέτει να έχει γίνει ταξινόμηση της ακολουθίας

$$\text{δηλαδή } a_1 \leq a_2 \leq \dots \leq a_n$$

Κάθε βήμα είτε βρίσκει το v είτε αποκλείει τα μισά σχεδόν στοιχεία της ακολουθίας που απομένουν.

θέση	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
στοιχείο	3	8	9	12	28	33	45	56	88	90	102	128	233	245	356

$v = 111$

↑

$$\text{θέση μέσου στοιχείου} = (9+11)/2 = 10 \quad a_{10} = 90 < v$$

Παράδειγμα: Δυαδική Αναζήτηση

Μας δίνεται μία ακολουθία από n ακέραιους $A = a_1, a_2, \dots, a_n$
και ένας ακέραιος v

Θέλουμε να βρούμε ένα $a_j \in A$ τέτοιο ώστε $a_j = v$

Δυαδική αναζήτηση: προϋποθέτει να έχει γίνει ταξινόμηση της ακολουθίας

$$\text{δηλαδή } a_1 \leq a_2 \leq \dots \leq a_n$$

Κάθε βήμα είτε βρίσκει το v είτε αποκλείει τα μισά σχεδόν στοιχεία της ακολουθίας που απομένουν.

θέση	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
στοιχείο	3	8	9	12	28	33	45	56	88	90	102	128	233	245	356

↑

$v = 111$

θέση μέσου στοιχείου = $(11+11)/2 = 11$ $a_{11} = 102 < v$

Παράδειγμα: Δυαδική Αναζήτηση

Μας δίνεται μία ακολουθία από n ακέραιους $A = a_1, a_2, \dots, a_n$
και ένας ακέραιος v

Θέλουμε να βρούμε ένα $a_j \in A$ τέτοιο ώστε $a_j = v$

Δυαδική αναζήτηση: προϋποθέτει να έχει γίνει ταξινόμηση της ακολουθίας

$$\text{δηλαδή } a_1 \leq a_2 \leq \dots \leq a_n$$

Κάθε βήμα είτε βρίσκει το v είτε αποκλείει τα μισά σχεδόν στοιχεία της ακολουθίας που απομένουν.

θέση	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
στοιχείο	3	8	9	12	28	33	45	56	88	90	102	128	233	245	356

$v = 111$

↑

$$\text{θέση μέσου στοιχείου} = (11+11)/2 = 11 \quad a_{11} = 102 < v$$

Παράδειγμα: Δυαδική Αναζήτηση

Μας δίνεται μία ακολουθία από n ακέραιους $A = a_1, a_2, \dots, a_n$
και ένας ακέραιος v

Θέλουμε να βρούμε ένα $a_j \in A$ τέτοιο ώστε $a_j = v$

Δυαδική αναζήτηση: προϋποθέτει να έχει γίνει ταξινόμηση της ακολουθίας

$$\text{δηλαδή } a_1 \leq a_2 \leq \dots \leq a_n$$

Κάθε βήμα είτε βρίσκει το v είτε αποκλείει τα μισά σχεδόν στοιχεία της ακολουθίας που απομένουν.

θέση	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
στοιχείο	3	8	9	12	28	33	45	56	88	90	102	128	233	245	356

↑

Το v δεν υπάρχει στην ακολουθία. Βρήκαμε όμως το αμέσως μικρότερο στοιχείο.

Παράδειγμα: Δυαδική Αναζήτηση

Μας δίνεται μία ακολουθία από n ακέραιους $A = a_1, a_2, \dots, a_n$
και ένας ακέραιος v

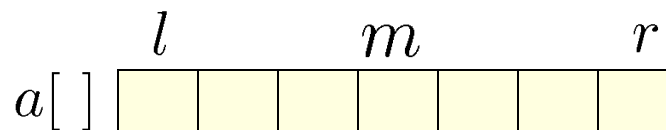
Θέλουμε να βρούμε ένα $a_j \in A$ τέτοιο ώστε $a_j = v$

Δυαδική αναζήτηση: προϋποθέτει να έχει γίνει ταξινόμηση της ακολουθίας

$$\text{δηλαδή } a_1 \leq a_2 \leq \dots \leq a_n$$

Κάθε βήμα είτε βρίσκει το v είτε αποκλείει τα μισά σχεδόν στοιχεία της ακολουθίας που απομένουν.

```
public static int binarySearch(int a[], int v, int l, int r)
{
    while (r >= l)
    {
        int m = (l+r)/2;  ←  $m = \left\lfloor \frac{l+r}{2} \right\rfloor$ 
        if (v == a[m]) return m;
        if (v < a[m]) r = m-1; else l = m+1;
    }
    return -1;
}
```



Παράδειγμα: Δυαδική Αναζήτηση

Μας δίνεται μία ακολουθία από n ακέραιους $A = a_1, a_2, \dots, a_n$
και ένας ακέραιος v

Θέλουμε να βρούμε ένα $a_j \in A$ τέτοιο ώστε $a_j = v$

Δυαδική αναζήτηση: προϋποθέτει να έχει γίνει ταξινόμηση της ακολουθίας

$$\text{δηλαδή } a_1 \leq a_2 \leq \dots \leq a_n$$

Κάθε βήμα είτε βρίσκει το v είτε αποκλείει τα μισά σχεδόν στοιχεία της ακολουθίας που απομένουν.

Ανάλυση: Έστω $T(n)$ ο χρόνος της αναζήτησης σε ακολουθία n στοιχείων.

Παράδειγμα: Δυαδική Αναζήτηση

Μας δίνεται μία ακολουθία από n ακέραιους $A = a_1, a_2, \dots, a_n$
και ένας ακέραιος v

Θέλουμε να βρούμε ένα $a_j \in A$ τέτοιο ώστε $a_j = v$

Δυαδική αναζήτηση: προϋποθέτει να έχει γίνει ταξινόμηση της ακολουθίας

$$\text{δηλαδή } a_1 \leq a_2 \leq \dots \leq a_n$$

Κάθε βήμα είτε βρίσκει το v είτε αποκλείει τα μισά σχεδόν στοιχεία της ακολουθίας που απομένουν.

Ανάλυση: Έστω $T(n)$ ο χρόνος της αναζήτησης σε ακολουθία n στοιχείων.

$$\begin{aligned} \text{Ισχύει } T(n) &= T(n/2) + \Theta(1) = T(n/2) + c && \text{χειρότερη περίπτωση} \\ &= T(n/4) + 2c = T(n/8) + 3c \\ &= \dots = c \lg n = \Theta(\lg n) \end{aligned}$$

Αναζήτηση

Μας δίνεται μία ακολουθία από n ακέραιους $A = a_1, a_2, \dots, a_n$
και ένας ακέραιος v

Θέλουμε να βρούμε ένα $a_j \in A$ τέτοιο ώστε $a_j = v$

⇒ Χρόνος δυαδικής αναζήτησης = $\Theta(\log n)$

⇒ Με πίνακες διασποράς επιτυγχάνουμε $\Theta(1)$ χρόνο.
(Προσεχώς σε μερικές εβδομάδες...)

Αναζήτηση

Εύρεση προκατόχου

Μας δίνεται μία ακολουθία από n ακέραιους $A = a_1, a_2, \dots, a_n$
και ένας ακέραιος v

Θέλουμε να βρούμε τον μεγαλύτερο $a_j \in A$ τέτοιο ώστε $a_j \leq v$

⇒ Χρόνος δυαδικής αναζήτησης = $\Theta(\log n)$

Καλύτερος δυνατός ασυμπτωτικός χρόνος όταν βασιζόμαστε μόνο σε πράξεις σύγκρισης ($>$, $<$, $=$).

⇒ Με πιο περίπλοκες δομές μπορούμε να πετύχουμε $O(\log \log n)$ χρόνο.

Αναδρομικές σχέσεις

Πολύ συχνά ο χρόνος εκτέλεσης ενός αλγόριθμου μπορεί να εκφραστεί μέσω κάποιας αναδρομικής σχέσης.

Παράδειγμα: Ακολουθιακή αναζήτηση και απαλοιφή ενός στοιχείου μέχρι να μείνει κενή ακολουθία.

128 233 88 9 12 45 33 28 56 8 90 102 3 245 356


= 45?

Αναδρομικές σχέσεις

Πολύ συχνά ο χρόνος εκτέλεσης ενός αλγόριθμου μπορεί να εκφραστεί μέσω κάποιας αναδρομικής σχέσης.

Παράδειγμα: Ακολουθιακή αναζήτηση και απαλοιφή ενός στοιχείου μέχρι να μείνει κενή ακολουθία.

128 233 88 9 12 ~~45~~ 33 28 56 8 90 102 3 245 356

↑
= 45?

Αναδρομικές σχέσεις

Πολύ συχνά ο χρόνος εκτέλεσης ενός αλγόριθμου μπορεί να εκφραστεί μέσω κάποιας αναδρομικής σχέσης.

Παράδειγμα: Ακολουθιακή αναζήτηση και απαλοιφή ενός στοιχείου μέχρι να μείνει κενή ακολουθία.

128 233 88 9 12 33 28 56 8 90 102 3 245 356
↑
= 8?

Αναδρομικές σχέσεις

Πολύ συχνά ο χρόνος εκτέλεσης ενός αλγόριθμου μπορεί να εκφραστεί μέσω κάποιας αναδρομικής σχέσης.

Παράδειγμα: Ακολουθιακή αναζήτηση και απαλοιφή ενός στοιχείου μέχρι να μείνει κενή ακολουθία.

Έστω $T(n)$ ο χρόνος εκτέλεσης για ακολουθία n στοιχείων στη **χειρότερη περίπτωση**.

$$\text{Ισχύει } T(n) = T(n - 1) + n, n \geq 2$$

$$T(n) = 1, n = 1$$

Αναδρομικές σχέσεις

Πολύ συχνά ο χρόνος εκτέλεσης ενός αλγόριθμου μπορεί να εκφραστεί μέσω κάποιας αναδρομικής σχέσης.

Παράδειγμα: Ακολουθιακή αναζήτηση και απαλοιφή ενός στοιχείου μέχρι να μείνει κενή ακολουθία.

Έστω $T(n)$ ο χρόνος εκτέλεσης για ακολουθία n στοιχείων στη **χειρότερη περίπτωση**.

$$\text{Ισχύει } T(n) = T(n - 1) + n, n \geq 2$$

$$T(n) = 1, n = 1$$

Με τη **μέθοδο της αντικατάστασης** λαμβάνουμε

$$T(n) = T(n - 1) + n = T(n - 2) + (n - 1) + n = \dots$$

Αναδρομικές σχέσεις

Παράδειγμα: Ακολουθιακή αναζήτηση και απαλοιφή ενός στοιχείου μέχρι να μείνει κενή ακολουθία.

Έστω $T(n)$ ο χρόνος εκτέλεσης για ακολουθία n στοιχείων στη **χειρότερη περίπτωση**.

$$\text{Ισχύει } T(n) = T(n - 1) + n, n \geq 2$$

$$T(n) = 1, n = 1$$

Με τη **μέθοδο της αντικατάστασης** λαμβάνουμε

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

Αναδρομικές σχέσεις

Παράδειγμα: Ακολουθιακή αναζήτηση και απαλοιφή ενός στοιχείου μέχρι να μείνει κενή ακολουθία.

Έστω $T(n)$ ο χρόνος εκτέλεσης για ακολουθία n στοιχείων στη **χειρότερη περίπτωση**.

$$\begin{aligned}\text{Ισχύει } T(n) &= T(n-1) + n, n \geq 2 \\ T(n) &= 1, n = 1\end{aligned}$$

Με **επαγωγή**: Θέλουμε να δείξουμε ότι $T(n) = O(n^2)$

δηλαδή ότι $T(n) \leq cn^2$ για κάποια θετική σταθερά c .

Η βάση της επαγωγής είναι $T(1) = 1 \leq c1^2 = c$ που ισχύει για $c \geq 1$

Επαγωγικό βήμα: Έστω $T(n-1) \leq c(n-1)^2$

$$\begin{aligned}\text{Έχουμε } T(n) &= T(n-1) + n \leq c(n-1)^2 + n \leq c[(n-1)^2 + n] \\ &= c(n^2 - n + 1) \leq cn^2, n \geq 1\end{aligned}$$

Αναδρομικές σχέσεις

Παράδειγμα: Ακολουθιακή αναζήτηση και απαλοιφή ενός στοιχείου μέχρι να μείνει κενή ακολουθία.

Έστω $T(n)$ ο χρόνος εκτέλεσης για ακολουθία n στοιχείων στη **χειρότερη περίπτωση**.

$$\text{Ισχύει } T(n) = T(n-1) + n, n \geq 2$$

$$T(n) = 1, n = 1$$

Με **επαγωγή**: Επίσης ισχύει $T(n) = \Omega(n^2)$. Θα δείξουμε επαγωγικά ότι $T(n) \geq c'n^2$ για κάποια θετική σταθερά c' .

Η βάση της επαγωγής είναι $T(1) = 1 \geq c'1^2 = c'$ που ισχύει για $c' \leq 1$

Επαγωγικό βήμα: Έστω $T(n-1) \geq c'(n-1)^2$

$$\begin{aligned} \text{Έχουμε } T(n) &= T(n-1) + n \geq c'(n-1)^2 + n = c'n^2 - 2c'n + c' + n \\ &\geq c'n^2 + (1 - 2c')n \geq c'n^2 \quad \text{που ισχύει για } c' \leq 1/2 \end{aligned}$$

Αναδρομικές σχέσεις

Παράδειγμα: Ακολουθιακή αναζήτηση και απαλοιφή ενός στοιχείου μέχρι να μείνει κενή ακολουθία.

Έστω $T(n)$ ο χρόνος εκτέλεσης για ακολουθία n στοιχείων στη **χειρότερη περίπτωση**.

$$\text{Ισχύει } T(n) = T(n - 1) + n, n \geq 2$$

$$T(n) = 1, n = 1$$

$$\text{Άρα έχουμε } T(n) = O(n^2) \text{ και } T(n) = \Omega(n^2)$$

$$\Rightarrow T(n) = \Theta(n^2)$$

Αναδρομικές σχέσεις

Παράδειγμα: $T(n) = 2T(n/2) + n, n \geq 2$
 $T(n) = 0, n = 1$

Με τη μέθοδο της αντικατάστασης λαμβάνουμε

$$\begin{aligned} T(n) &= 2T(n/2) + n = 2[2T(n/4) + n/2] + n \\ &= 4T(n/4) + 2n = 4[2T(n/8) + n/4] + 2n \\ &= 8T(n/8) + 3n = \dots \end{aligned}$$

Αναδρομικές σχέσεις

Παράδειγμα: $T(n) = 2T(n/2) + n, n \geq 2$
 $T(n) = 0, n = 1$

Με τη μέθοδο της αντικατάστασης λαμβάνουμε

$$\begin{aligned} T(n) &= 2T(n/2) + n = 2[2T(n/4) + n/2] + n \\ &= 4T(n/4) + 2n = 4[2T(n/8) + n/4] + 2n \\ &= 8T(n/8) + 3n = \dots \\ &= 2^k T(n/2^k) + kn = \Theta(kn) = \Theta(n \log n) \end{aligned}$$

Αναδρομικές σχέσεις

Παράδειγμα: $T(n) = 2T(n/2) + n, n \geq 2$
 $T(n) = 0, n = 1$

Εναλλακτικά έχουμε: $T(2^n) = 2T(2^{n-1}) + 2^n$

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

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

οπότε $T(2^n) = n2^n \Rightarrow T(n) = \Theta(n \log n)$