

Master Theorem

$$T(n) = aT\left(\frac{n}{b}\right) + f(n), \quad a \neq 1, \quad b > 1$$

Αν πρέπει να το εφαρμόζουμε, πρέπει να φέρω το $f(n)$ στην ίδια μορφή.

(π.χ. $T(n) = 3T\left(\frac{n}{2}\right) + n$ $a=3, \quad b=2, \quad f(n)=n$)

Έχουμε 3 περιπτώσεις

- i) Αν $f(n) = O(n^{\log_b a - \epsilon})$, τότε $T(n) = \theta(n^{\log_b a})$
- ii) Αν $f(n) = \theta(n^{\log_b a} \log^k n)$, $k \neq 0$, τότε $T(n) = \theta(n^{\log_b a} \log^{k+1} n)$
- iii) Αν $f(n) = \Omega(n^{\log_b a + \epsilon})$, τότε $T(n) = \theta(f(n))$

Ασκήσεις

• $T(n) = 3T\left(\frac{n}{2}\right) + n^2$

$a=3, \quad b=2, \quad f(n)=n^2$

$n^2 = n^{\log_2 3 + 1 - \epsilon}$ από iii) $\Rightarrow T(n) = \theta(n^2)$

• $T(n) = 4T\left(\frac{n}{2}\right) + n^2$

$a=4, \quad b=2$ και $f(n)=n^2$

$n^2 = n^{\log_2 4}$, από $T(n) = \theta(n^{\log_2 4} \cdot \log n)$, έχουμε $k=0$

Υπάρχει περίπτωση να μην ανήκει και σε κάποια από τις 3 περιπτώσεις.

$$2^n = n^{\log_2 1 + 1}$$

- $T(n) = T\left(\frac{n}{2}\right) + 2^n$

$$a=1, b=2 \quad f(n)=2^n$$

$$2^n = n^{\log_2(1+1)} \quad \text{Είπαμε ότι } 3^\circ \text{ περίπτωση}$$

• Απόδειξη: να βρω ένα μικρό αριθμό α : $2^n > n$
 $T(n) = \Theta(2^n)$

- $T(n) = 16T\left(\frac{n}{4}\right) + n$

$$a=16 \quad b=4 \quad f(n)=n$$

$$n = n^{\log_4 16 - \epsilon} \quad T(n) = \Theta(n^{\log_4 16})$$

- $T(n) = 2T\left(\frac{n}{2}\right) + n \log n$

$$a=2, b=2 \quad f(n)=n \log n$$

$$n \log n = n^{\log_2 2} \cdot \log n \quad \text{αρα, } T(n) = \Theta(n^{\log_2 2} \log^2 n)$$

Άσκηση i) $T(n) = 2T\left(\frac{n}{2}\right) + \frac{n}{\log n}$

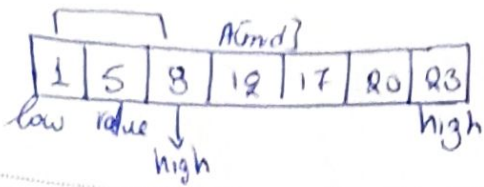
iv) $T(n) = 64T\left(\frac{n}{8}\right) - n^2 \log n$

ii) $T(n) = 16T\left(\frac{n}{4}\right) + n!$

v) $T(n) = 7T\left(\frac{n}{3}\right) + n^2$

iii) $T(n) = 6T\left(\frac{n}{3}\right) + n^2 \log n$

vi) $T(n) = 4T\left(\frac{n}{2}\right) + \log n$



```

int BinarySearch (Type A[0, ..., N-1], value, low, high) {
    if (high < low)
        return -1;
    mid = low + (high - low) / 2;
    if (A[mid] > value)
        return BinarySearch(A, value, low, mid - 1);
    else if (A[mid] < value)
        return BinarySearch(A, value, mid + 1, high);
    else
        return mid;
}

```

Ασκηση 3: 1) Παρουσιάστε σύντομη περιγραφή του τρόπου λειτουργίας του αλγορίθμου.

2) Ισχυλάστε τον BinarySearch (A, 14, 0, 9) όπου ο πίνακας A είναι ο $A = [2, 4, 6, 8, 10, 12, 14, 16, 18, 20]$

3) Παρουσιάστε αναδρομική σχέση που να περιγράφει τη χρονική πολυπλοκότητα $T(n)$ του αλγορίθμου.

4) Τι τμήμα είναι η πολυπλοκότητα του BinarySearch; Αποδείξτε τον ισχυρισμό.