Recursion (Özyineleme)

Factorial of a number

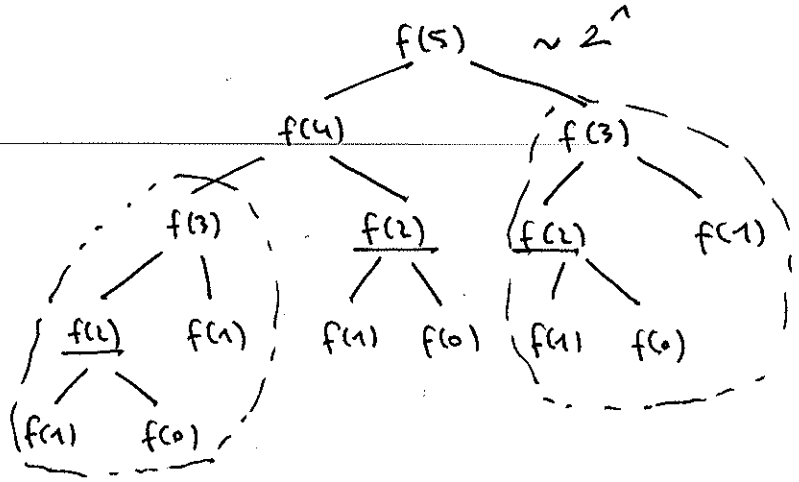
$f(n) = 1.2. \_\_ .n$ → non-recursive factorial

$$f(n) = \begin{cases} 1 & \text{if } n=0 \\ n.f(n-1) & \text{otherwise} \end{cases}$$ → recursive factorial

Fibonacci Numbers

$f_0 = f_1 = 1$

$f_n = f_{n-1} + f_{n-2}$, $n > 1$



stack of degree depth n is needed

Recursive

- in efficient
  too many redundant operations

non-recursive

- very efficient ~ n. constant

```
tn1 = 1
tn2 = 1
for 1 → n
    tn = tn1 + tn2
    tn2 = tn1
    tn1 = tn
end
```

Recurrence Relation (Özyineli ilişki)

* Eğer bir algoritma kendine özyineli bir çağrı içeriyorsa - bu algoritmanın çalışma süresi bir özyineli denklem ile açıklanır.
* Özyineleme: Bir fonksiyonu daha küçük girdilere verdiği değerler cinsinden açıklayan bir denklem ve ya eşitsizliktir.

Tanım: $f: \mathbb{N} \to \mathbb{R}$ fonksiyonu, pozitif n_0 gibi bir tamsayı için aşağıdaki kuralları sağlıyorsa özyineli ilişki söz konusudur:

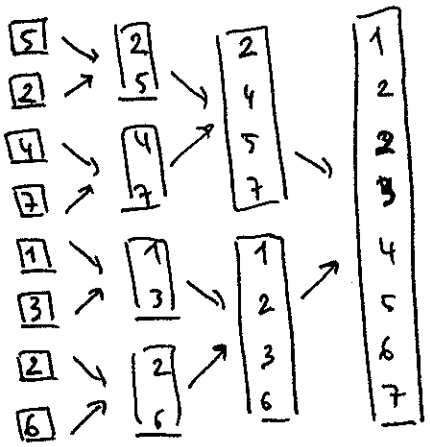
i) $f(1), f(2), \ldots, f(n_0)$ değerleri bilinmekte

ii) $n > n_0$ olmak üzere $f(n)$, $f(1), f(2), \ldots, f(n-1)$ değerleri ile açıklanabilir.

Özyineli ilişkilerin gösteriminde amaç n. terim için açık bir formül bulmaktır.

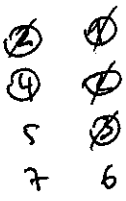
Örnek: MERGE_SORT

$T(n)$	merge sort. $A[1 \ldots n]$
$\Theta(1)$	1. if $n=1$, done
	2. recursively sort
$2T(n/2)$	$A[1 \ldots \lceil n/2 \rceil]$ and $A[\lceil n/2 \rceil + 1, \ldots, n]$
$\Theta(n)$	3. merge 2 sorted lists



ilk liste (sıralı değil)

altrutin merge:



1 2 2 3 ...

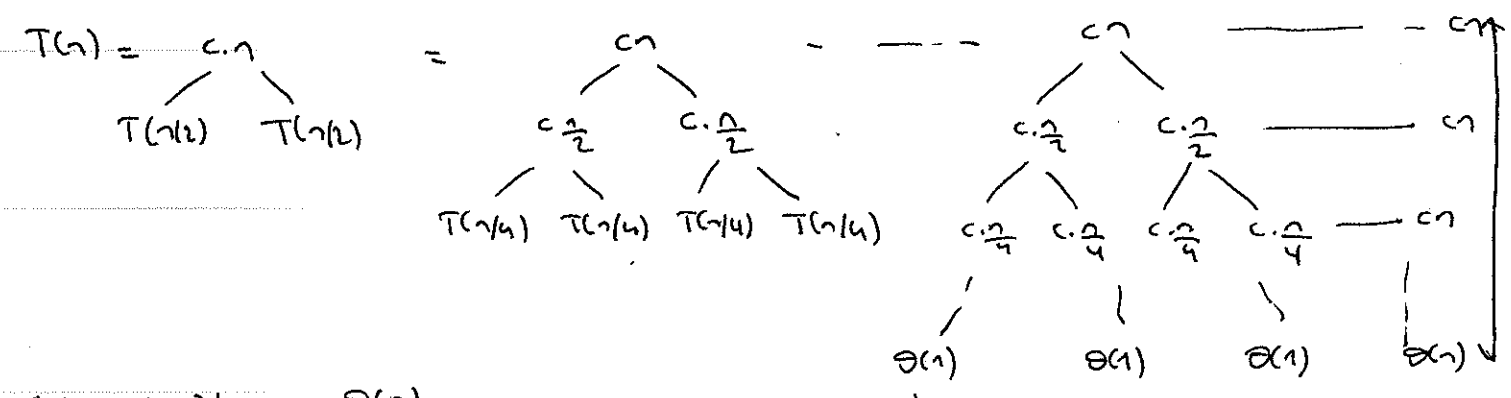
time = $\Theta(n)$

toplam n eleman varsa

(doğrusal zaman)

$$T(n) = \begin{cases} \Theta(1) & \text{if } n=1 \\ 2T(n/2) + \Theta(n) & \text{if } n>1 \end{cases}$$

Recursion tree $T(n) = 2T(n/2) + c \cdot n$, $c>0$



total $= (c \cdot n) \log n + \Theta(n)$

$= \Theta(n \cdot \log n)$

# of leaves $= 2^h = n$

Solving recurrence relations

1. Iteration method

recurrence → sum

ex: $T(n) = T(n-1) + n$, $n \geq 2$ s.t $T(1) = 1$

↳ a recursive program loops through the input to eliminate one item, and recursively is called each time

$$T(n) = T(n-1) + n$$
$$= T(n-2) + (n-1) + n$$
$$= T(n-3) + (n-2) + (n-1) + n$$
$$\vdots$$
$$= T(1) + 2 + 3 + \dots + (n-1) + n$$
$$= 1 + 2 + 3 + \dots + n$$
$$= \frac{n \cdot (n+1)}{2} \in O(n^2)$$

ex: $T(n) = t_n = 2\, t_{n/2} + n$, $n \geq 2$ s.t $t_1 = 0$

$$t_n = 2\, t_{n/2} + n$$
$$= 2 \cdot (2\, t_{n/4} + n/2) + n = 4\, t_{n/4} + 2n = 2^2\, t_{n/4} + 2 \cdot n$$
$$= 4\,(2 \cdot t_{n/8} + \tfrac{n}{4}) + 2n = 8 \cdot t_{n/8} + 3n = 2^3\, t_{n/8} + 3 \cdot n$$

Let $n = 2^k$ (i.e. $k = \log n$)

$$t_n = 2^k\, t_{k/k} + n \cdot \overbrace{(1+1+\dots+1)}^{k}$$
$$= 2^k\, t_1 + n \cdot k$$
$$\rightarrow t_n = n \cdot (\log n) \in O(n \log n)$$

2. Substitution Method

a. Predict the solution

b. Use induction to verify and proof the solution

(c. Find the constants)

ex. $t(1) = 0$

$t(n) = 1 + t(n/2)$

a. Predict $t(n) = \log_2 n$

b. Base : $t(1) = \log_2 1 = 0$ ✓

Inductive step : $t(N) = \log_2 N$ ← inductive hypothesis

$$t(N+1) = 1 + \log_2[(N+1)/2]$$

$$= 1 + (\log_2(N+1) - \log_2 2)$$

$$= \log_2(N+1)$$ ✓

ex. $T(1) = \Theta(1)$

$T(n) = 4\,T(n/2) + n$

Estimate the upper bound! (as tight as possible)

Recall: The upper bound of $f(n)$ is $g(n)$ if there exists c_0 & N such that for all $n > N$:

$$c_0 g(n) \geq f(n) \quad (\text{or } f(n) \leq c \cdot g(n))$$

which is shown as $f(n) = O(g(n))$

1st prediction: $T(n) = O(n^3)$

Inductive step: Assume $T(k) \leq c \cdot k^3$ for $k < n$

$$T(n) = 4 \cdot T(n/2) + n$$

$$\leq 4 \cdot c \cdot \left(\frac{n}{2}\right)^3 + n$$

$$= \frac{1}{2} \cdot c \cdot n^3 + n$$

$$= c \cdot n^3 - \left(\frac{1}{2} c \cdot n^3 - n\right)$$

$$\leq c \cdot n^3 \quad \text{if } \frac{1}{2} \cdot c \cdot n^3 - n \geq 0$$

e.g. $c \geq 1$, $n \geq 1$

Base: $T(1) = \Theta(1) \leq c$

if c is chosen sufficiently large.

2nd prediction: $T(n) = O(n^2)$

Inductive step: Assume $T(k) \leq c \cdot k^2$ for $k < n$

$$T(n) = 4 \cdot T(n/2) + n$$
$$\leq 4 \cdot c \cdot \left(\frac{n}{2}\right)^2 + n$$
$$= cn^2 + n$$
$$= cn^2 - \underbrace{(-n)}_{\geq 0} \quad \times$$
$$\leq cn^2$$

3rd prediction: may be the lower order terms important!

Inductive step: Assume $T(k) \leq c_1 k^2 - c_2 k$ for $k < n$

$$T(n) = 4\, T(n/2) + n$$
$$\leq 4 \cdot \left[c_1 \left(\frac{n}{2}\right)^2 - c_2 \cdot \left(\frac{n}{2}\right)\right] + n$$
$$= c_1 n^2 + (1 - 2c_2) n$$
$$= c_1 n^2 - c_2 \cdot n - \underbrace{(-1 + c_2) \cdot n}_{\overset{!}{\geq} 0}$$

$c_2 - 1 \geq 0$ if $c_2 \geq 1$

$$\leq c_1 n^2 - c_2 n \quad \text{if } c_2 \geq 1$$

Base: $T(1) \leq c_1 - c_2$

$\in \Theta(1)$ if c_1 sufficiently large wrt. c_2 ($c_1 \gg c_2$)

- Try to solve
  - $t(n) = t(n/2) + n = O(n)$
  - $t(n) = 2 + t(n/2) + n = O(n \log n)$

3. Solving Recurrences using Characteristic polynomial

recurrence as dynamical systems

↳ solving dynamical systems

$$T(k+n) + a_{n-1} T(k+n-1) + \cdots + a_0 (T(n)) = f(n)$$

↳ linear discrete-time dynamical system (difference equation)

when $f(n) = 0$ ⇒ homogeneous eqn.

General structure of the solutions: $T(k) = T_h(k) + T_p(k)$

↓ homogeneous solution ↓ partial solution

Homogeneous Equations

$$T(k+n) + a_{n-1} T(k+n-1) + \cdots + a_0 T(n) = 0$$

prediction: Let $T(k) = \lambda^k$, λ: scalar

$$\lambda^{k+n} + a_{n-1} \lambda^{k+n-1} + \cdots + a_0 \lambda^k = 0$$

$$\lambda^k \underbrace{(\lambda^n + a_{n-1} \lambda^{n-1} + \cdots + a_0)}_{n^{th}\text{-order polynomial}} = 0$$

$\boxed{p(\lambda) = 0}$: characteristic polynomial

* Let us assume $p(\lambda)$ have n different roots $\lambda_1, \lambda_2, \cdots, \lambda_n$

$$T_h(k) = \sum_{i=1}^{n} \alpha_i (\lambda_i)^k \qquad \{\alpha_i\}_1^n \text{ is estimated using initial conditions}$$

ex: Fibonacci series

$$F(k) = F(k-1) + F(k-2), \quad F(0) = 0, F(1) = 1$$

$$\lambda^2 - \lambda - 1 = 0 \Rightarrow \lambda = \frac{1 \mp \sqrt{5}}{2} \qquad * \text{Recall } \lambda = \frac{-b \mp \sqrt{b^2 - 4ac}}{2a}$$

$$\lambda_1 \approx 1.618$$

$$\lambda_2 \approx -0.618$$

$$F(k) = \alpha_1 \left(\frac{1+\sqrt{5}}{2}\right)^k + \alpha_2 \left(\frac{1-\sqrt{5}}{2}\right)^k \quad \text{Find } \alpha_1, \alpha_2 \text{ using } F(0)=0, F(1)=1$$

for $k=0$ $\quad \alpha_1 + \alpha_2 = 0$

for $k=1$ $\quad \alpha_1 \left(\frac{1+\sqrt{5}}{2}\right) + \alpha_2 \left(\frac{1-\sqrt{5}}{2}\right) = 1$

$\Rightarrow \quad \alpha_1 = \frac{1}{\sqrt{5}}, \quad \alpha_2 = -\frac{1}{\sqrt{5}} \quad \Rightarrow \quad F(k) \in O\left(\left(\frac{1+\sqrt{5}}{2}\right)^k\right)$

ex: $t_n - 3t_{n-1} - 4t_{n-2} = 0 \quad , \quad n \geq 2 \quad$ s.t. $t_0 = 0, t_1 = 1$

$\lambda^2 - 3\lambda - 4 = 0$

$\Rightarrow \quad (\lambda - 4)(\lambda + 1) = 0$

$\lambda_1 = -1$

$\lambda_2 = 4$

$t_n = \alpha_1 (-1)^n + \alpha_2 (4)^n$

I.C. $\Rightarrow$ $\left.\begin{aligned} \alpha_1 + \alpha_2 &= 0 \\ -\alpha_1 + 4\alpha_2 &= 1 \end{aligned}\right\} \quad \alpha_1 = -\frac{1}{5}, \quad \alpha_2 = \frac{1}{5}$

$\Rightarrow t_n = \frac{1}{5}\left[4^n - (-1)^n\right] \in O(4^n)$

ex: $t_{n+2} - 2a t_{n+1} + a^2 t_n = 0$

$\lambda^2 - 2a\lambda + a^2 \Rightarrow (\lambda - a)^2 \quad \Rightarrow \quad \lambda_1 = \lambda_2 = a$: repeated roots

$\Rightarrow t_n = \alpha_1 a^n + \alpha_2 . n . a^n \quad \in O(n.a^n)$

(For three repeated roots $\lambda^n, n.\lambda^n, n^2\lambda^n$)

ex: $t_n = 5t_{n-1} - 8t_{n-2} + 4t_{n-3}, \quad n \geq 3$ s.t $t_0 = 0, t_1 = 1, t_2 = 2$

$\lambda^3 - 5\lambda^2 + 8\lambda - 4 = 0$

$(\lambda - 1)(\lambda - 2)^2 = 0$

$\lambda_1 = 1, \quad \lambda_{2,3} = 2$

$\Rightarrow t_n = \alpha_1 1^n + \alpha_2 2^n + \alpha_3 n . 2^n$

I.C $\Rightarrow$ $\left.\begin{aligned} \alpha_1 + \alpha_2 &= 0 \\ \alpha_1 + 2\alpha_2 + 2\alpha_3 &= 1 \\ \alpha_1 + 4\alpha_2 + 8\alpha_3 &= 2 \end{aligned}\right| \begin{aligned} n &= 0 \\ n &= 1 \\ n &= 2 \end{aligned} \quad \Rightarrow \quad \begin{aligned} \alpha_1 &= -2 \\ \alpha_2 &= 2 \\ \alpha_3 &= -1/2 \end{aligned}$

$\rightarrow t_n = 2^{n+1} - n.2^{n-1} - 2 \quad \in \Theta(n.2^n)$

Inhomogeneous Equations

$$T(k+n) + a_{n-1} T(k+n-1) + \cdots + a_0 T(n) = f(n) \qquad (1)$$

solution $\Rightarrow$ $T(k) = T_h(k) + T_p(k)$

- $T_h(k)$: find considering $f(n) = 0$ $\rightarrow$ as in previous examples
- $T_p(k)$: requires exposure to similar solutions

Let us assume we find the partial solution $T_p(k)$

$$T_p(k+n) + a_{n-1} T_p(k+n-1) + \cdots + a_0 T_p(n) = f(n) \qquad (2)$$

Subtract (2) from (1)

$$T(k+n) - T_p(k+n) + a_{n-1}\left(T(k+n-1) - T_p(k+n-1)\right) + \cdots + a_0\left(T(n) - T_p(n)\right) = 0$$

change of variables $F(k) = T(k) - T_p(k)$ $(*)$

$\Rightarrow$ $F(k+n) + a_{n-1} F(k+n-1) + \cdots + a_0 F(n) = 0$ $\Rightarrow$ homogeneous eqn.

Find the explicit solution $F(k)$

From $(*)$, $\boxed{T(k) = F(k) + T_p(k)}$

- Note that finding the partial solution is not always easy!

Special case: $f(n) = b^n p(n)$

b : scalar

$p(n)$: d^{th}-order polynomial

$$T(k+n) + a_{n-1} T(k+n-1) + \cdots + a_0 T(n) = b^n p(n)$$

$$\Rightarrow \underbrace{\left(\lambda^n + a_{n-1}\lambda^{n-1} + \cdots + a_0\right)}_{\text{homogeneous}} \underbrace{(\lambda - b)^{d+1}}_{\text{partial}} = 0$$

ex: $a_n = 3a_{n-1} + n$, $n \geq 1$ s.t $a_0 = 1$

$f(n) = n = 1^n \cdot n$

$(\lambda^2 - 3\lambda)(\lambda - 1)^2 = 0$

$\Rightarrow a_n = c_1 3^n + \cancel{c_2 0^n} + c_3 1^n + c_4 \cdot n 1^n$

$= \underbrace{c_1 3^n}_{T_h(n)} + \underbrace{c_3 + c_4 n}_{T_p(n)}$

partial solution: $c_3 + c_4 n = 3.(\overbrace{c_3 + c_4 (n-1)}^{T_p(n-1)}) + n$

$= 3c_3 - 3c_4 + (1 + 3c_4) n$

$\Rightarrow (1)$ $c_3 = 3c_3 - 3c_4$

$c_4 = 1 + 3c_4$ $\Rightarrow c_3 = \frac{-3}{4}$, $c_4 = -\frac{1}{2}$ $\Rightarrow T_p(n) = -\frac{3}{4} + \left(\frac{-1}{2}\right) \cdot n$

homogeneous solution: $a_0 = 1 \Rightarrow c_1 (3)^0 - \frac{3}{4} - \frac{1}{2} \cdot (0) = c_1 - \frac{3}{4} = 1$

$\Rightarrow c_1 = \frac{7}{4}$

$\Rightarrow a_n = -\frac{3}{4} - \frac{1}{2} n + \frac{7}{4} (3)^n \in O(3^n)$

ex: $T(n) = 2T(n-1) + 3T(n-2) + 5^n$, $n \geq 2$, $T(0) = -2$, $T(1) = 1$

homogeneous solution: $\lambda^2 - 2\lambda - 3 = 0$

$(\lambda - 3)(\lambda + 1)$ $\lambda_1 = 3$

$\lambda_2 = -1$

partial solution: $f(n) = 1 \cdot 5^n \Rightarrow T_p(n) = c_1 \cdot 5^n$

$c_1 \cdot 5^n = 2 \cdot c_1 \cdot 5^{n-1} + 3 c_1 \cdot 5^{n-2} + 5^n$ (divide by 5^{n-2})

$25 c_1 = 10 c_1 + 3 c_1 + 25 \Rightarrow c_1 = \frac{25}{12} \Rightarrow T_p(n) = \frac{25}{12} 5^n$

$T_h(n) = c_2 (-1)^n + c_3 (3^n)$

$T(n) = c_2 (-1)^n + c_3 \cdot (3^n) + \frac{25}{12} (5^n)$

I.C. $T(0) = -2 = \frac{25}{12} + c_2 + c_3$

$T(1) = 1 = \frac{25}{12} \cdot 5 - c_2 + 3c_3$

$\Rightarrow c_2 = -\frac{17}{24}$, $c_3 = -\frac{27}{8}$

$\Rightarrow T(n) = -\frac{17}{24}(-1)^n - \frac{27}{8}(3^n) + \frac{25}{12}(5)^n \in O(5^n)$

ex: $T(n) = 4T(n/2) + n$, $n > 1$

change of variables , assume $n = 2^k$

$T(2^k) = 4T(2^{k-1}) + 2^k$

$S(k) = 4S(k-1) + 2^k$

$(\lambda - 4)(\lambda - 2)$

$\Rightarrow S(k) = c_1 \cdot 4^k + c_2 \cdot 2^k$

$\Rightarrow T(n) = c_1 n^2 + c_2 \cdot n \in O(n^2)$

- Try to solve
  - $T(n) = 3T(n/3) + n\log_3 n$, $n > 1$
  - $T(n) = 3T(n/2) + c \cdot n$
  - $T(n) = n \cdot T^2(n/2)$, $n > 1$ s.t. $T(1) = 6$