حل روابط بازگشتی — روش جایگذاری :

مثال برای روش جایگذاری: تابع بازگشتی زیر را که در یک فرم کلی داده شده است در نظر بگیرید:

$$T(n) = aT(n/b) + f(n)$$

گام ۱ جایگذاری →
$$\begin{aligned} T(n) &= aT(n/b) + f(n) \\ &= a\left(aT\left(\frac{n}{b^2}\right) + f(n/b)\right) + f(n) \\ &= a^2T\left(\frac{n}{b^2}\right) + af\left(\frac{n}{b}\right) + f(n) \end{aligned}$$

گام ۲ جایگذاری →
$$\begin{aligned} T(n) &= a^2T\left(\frac{n}{b^2}\right) + af(n/b) + f(n) \\ &= a^2\left(aT\left(\frac{n}{b^3}\right) + f\left(\frac{n}{b^2}\right)\right) + af\left(\frac{n}{b}\right) + f(n) \\ &= a^3T\left(\frac{n}{b^3}\right) + a^2f\left(\frac{n}{b^2}\right) + af(n/b) + f(n) \end{aligned}$$

$\vdots$

گام k جایگذاری →
$$T(n) = a^kT\left(\frac{n}{b^k}\right) + \sum_{i=0}^{k-1} a^i f\left(\frac{n}{b^i}\right)$$

تا کجا باید k را پیش ببریم؟ تا جایی که پایه رابطه بازگشتی = نمونه کوچکمان = $\frac{n}{b^k}$

برای مثال فرض کنید پایه رابطه بازگشتی 1 باشد، بنابراین داریم:

$$\frac{n}{b^k} = 1 \longrightarrow n = b^k \longrightarrow \log_b n = k$$

پس رابطه کلی زیر حاصل می‌شود:

$$T(n) = a^{\log_b n} . T(1) + \sum_{i=0}^{(\log_b n)-1} a^i f\left(\frac{n}{b^i}\right)$$

از آنجائیکه $a^{\log_b n} = n^{\log_b a}$ است، می‌توان عبارت بالا را به صورت زیر نیز بازنویسی کرد:

$$T(n) = n^{\log_b a} . T(1) + \sum_{i=0}^{(\log_b n)-1} a^i f\left(\frac{n}{b^i}\right)$$

در ادامه قصد داریم دو حالت خاص از فرم کلی رابطه بالا را در نظر گرفته و حل کنیم:

* فرض کنید $f(n) = c$ باشد، بنابراین داریم:

$$T(n) = n^{\log_b a} . T(1) + \underbrace{\sum_{i=0}^{(\log_b n)-1} c a^i}$$

$$\longrightarrow c \sum_{i=0}^{(\log_b n)-1} a^i \qquad k = (\log_b n) - 1$$

(*) نکته: برای هر عدد حقیقی چون $a \neq 1$ باشد، داریم

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

طبق (*) داریم

$$\longrightarrow c\left(\frac{a^{k+1} - 1}{a - 1}\right)$$

بنابراین:

$$T(n) = T(1) . n^{\log_b a} + \underbrace{c\left(\frac{a^{k+1} - 1}{a-1}\right)}_{O(n^{\log_b a})} = O(n^{\log_b a})$$

* فرض کنید $f(n)=n$ باشد، بنابراین داریم:

$$T(n)=n^{\log_b a}\cdot T(1)+\sum_{i=0}^{(\log_b n)-1} a^i f\left(\frac{n}{b^i}\right)$$

$$\sum_{i=0}^{(\log_b n)-1} a^i * \frac{n}{b^i}$$

$$= n\sum_{i=0}^{(\log_b n)-1}\left(\frac{a}{b}\right)^i$$

$$= n\,\frac{\left(\frac{a}{b}\right)^{(\log_b n)-1}-1}{\frac{a}{b}-1} = O\left(n^{(\log_b a)}\right)$$

نکته: اگر روابط بازگشتی پیچیده‌تر شود، مثلاً تابع بازگشتی براساس دو تابع کوچکتر تعریف شود:

$$T(n)=2T(n/2)+3T\left(\frac{2n}{5}\right)+n$$

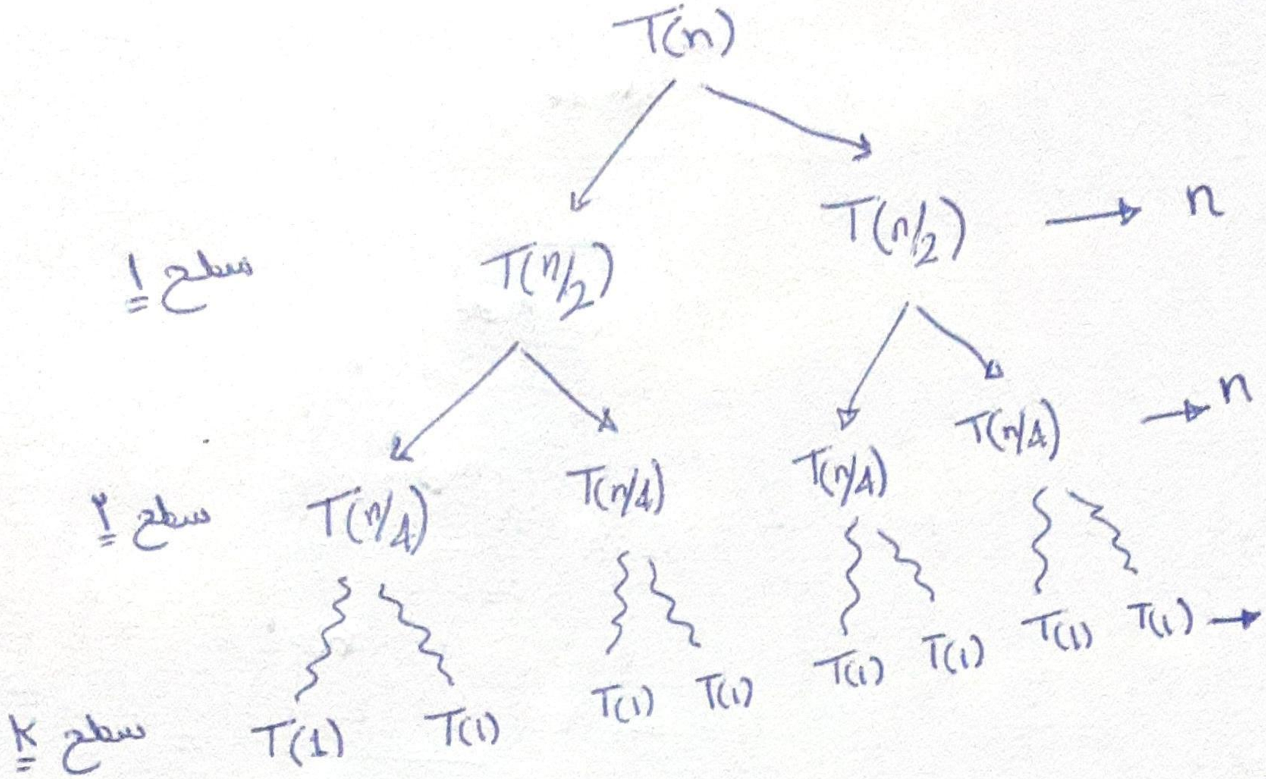
در این حالت روش جایگذاری آزاردهنده بوده و البته اکثر مواقع جواب ~~...~~ نمی‌دهد.

حل روابط بازگشتی - روش درخت بازگشت:

می‌توان برای حل روابط بازگشتی، درخت آن را رسم کرد که به آن درخت بازگشتی گفته می‌شود.

مثال):

$$T(n)=2T(n/2)+n$$



* تا کجا باید پیش بریم ←

$$\frac{n}{2^k} = 1 \Rightarrow 2^k = n \Rightarrow k = \log_2 n$$

* تعداد نودها در سطح k ←

$$2^k = 2^{\log_2 n} = n$$

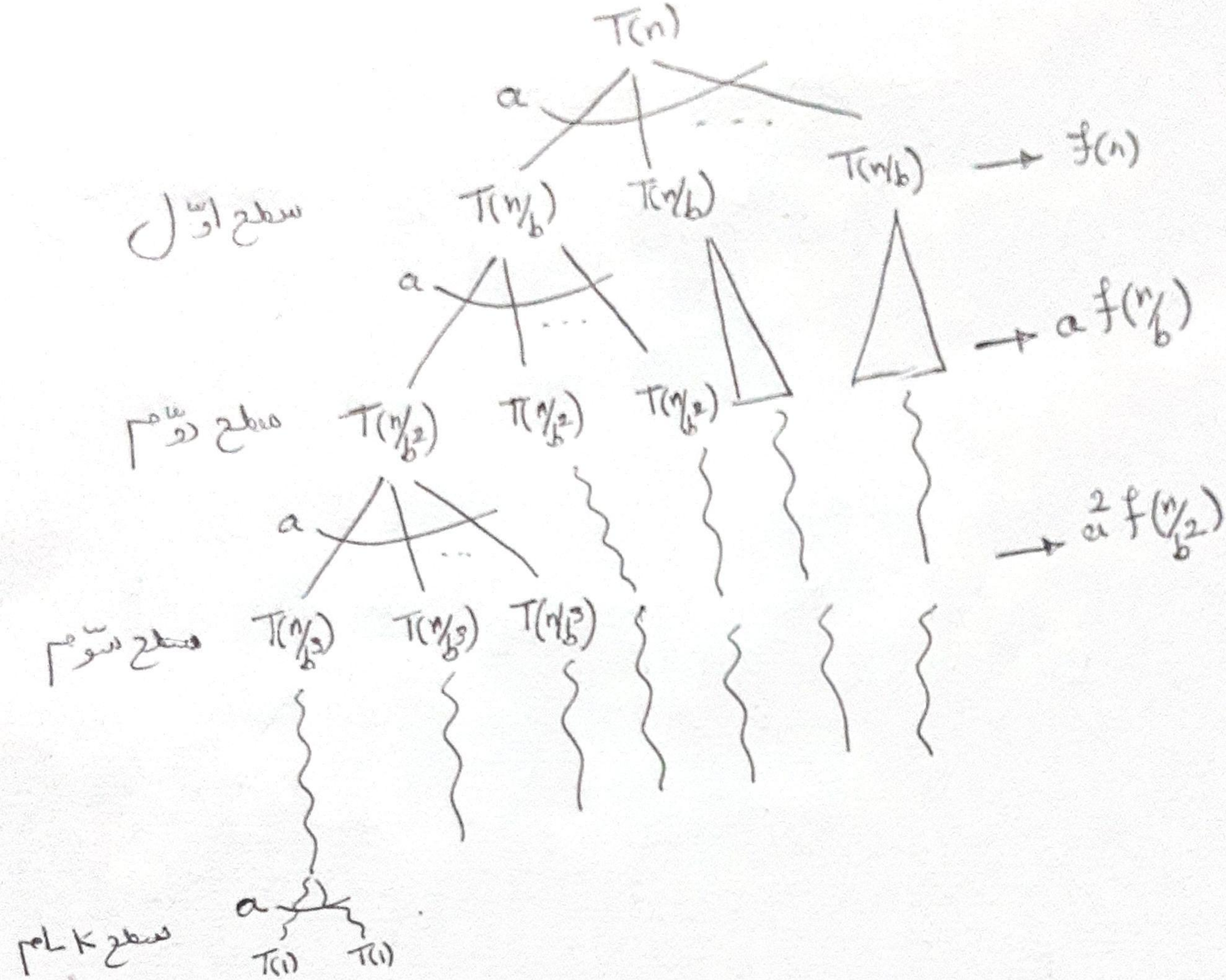
$$T(n) = \underbrace{\frac{n}{2}*2 + nT(1)}_{\text{سطح } k} + \sum_{i=1}^{k-1} n = n + nT(1) + (k-1)n$$

$$= n + nT(1) + n\log_2 n - n$$

$$= nT(1) + n\log_2 n = O(n\log_2 n)$$

مثال 2:

$$T(n) = aT(n/b) + f(n)$$



$$\frac{n}{b^k} = 1 \Rightarrow n = b^k \Rightarrow k = \log_b^n$$ ← تا کجا پیش می‌رویم

$$a^k = a^{\log_b^n} = n^{\log_b^a}$$ ← تعداد نودها در سطح k

$$f\left(\frac{n}{b^k}\right) = f\left(\frac{n}{b^{\log_b^n}}\right) = f\left(\frac{n}{n}\right) = f(1)$$ ← تابع f در مرحله k

$$T(n) = \underbrace{\left(n^{\log_b^a}\right) T(1) + a^{(\log_b^n)-1} f(1)}_{\text{سطح k}} + \underbrace{\sum_{i=1}^{k-1} a^{i-1} f\left(\frac{n}{b^{i-1}}\right)}_{\text{بقیه سطوح}}$$

$$\sum_{i=0}^{k-2} a^i f\left(\frac{n}{b^i}\right)$$ ← معادل