طراحی الگوریتم

درس دوم : رابطه های بازگشتی

مدرس:
فرشید شیرافکن

دانشجوی **دکتری دانشگاه تهران**
(کارشناسی و کارشناسی ارشد : کامپیوتر نرم افزار) (دکتری: بیو انفورماتیک)

رابطه های بازگشتی

یک رابطه ی بازگشتی برای تعداد ضرب‌ها در تابع فاکتوریل:

```
fact (n)
{
   if (n==0)
         return  1;
    else
         return   n*fact(n-1);
}
```

$$T(n)=\begin{cases}0 & n=0\\ 1+T(n-1) & n>=1\end{cases}$$

مثال

T(n) : تعداد ستاره های چاپ شده

```
void  f(int n)
 {
   if (n>=2)
    {
        f (n-1);
        print "***";
        f(n-2);
        print "*****";
        f(n-1);
     }
}
```

$$T(n) = 2T(n-1) + T(n-2) + 7$$

مسئله خرگوش ها

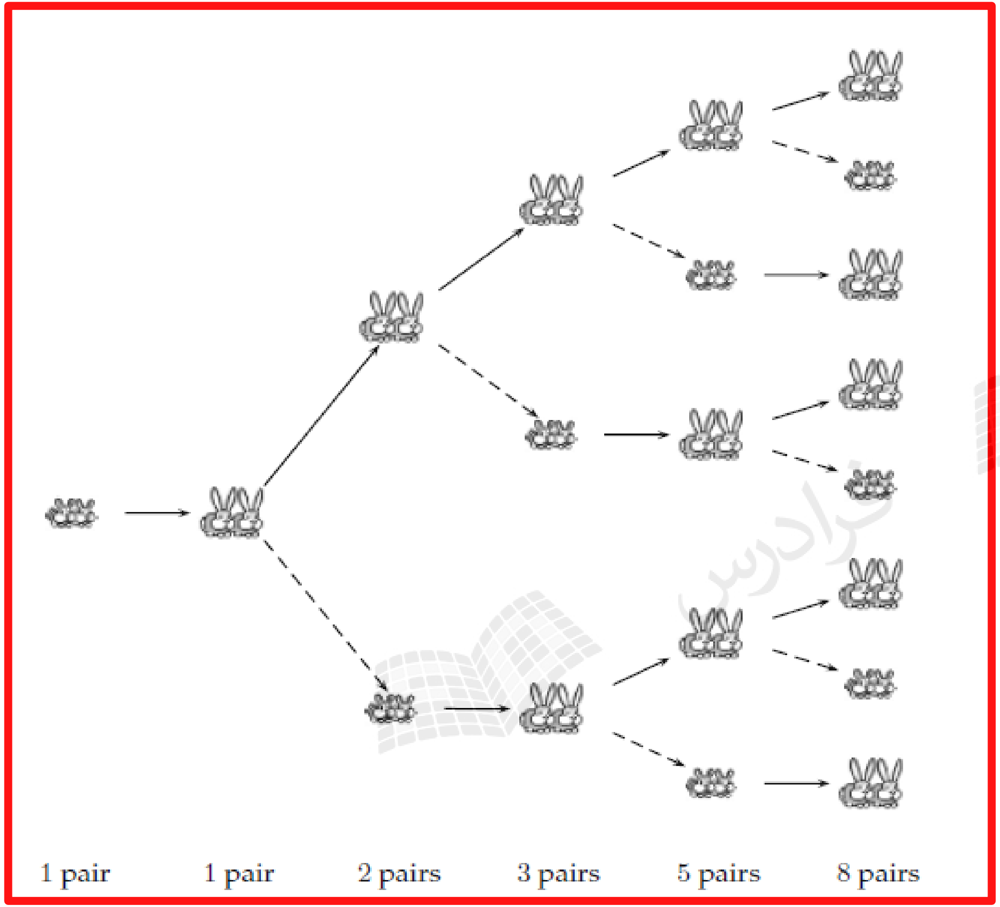
در یک جزیره یک جفت خرگوش نر و ماده نوزاد وجود دارد و مدل رشد جمعیت خرگوش ها به صورت زیر است:

(الف) خرگوش ها یک ماه پس از تولد به سن بلوغ می رسند.

(ب) یک ماه پس از رسیدن به سن بلوغ و از آن به بعد همه ماهه هر جفت خرگوش بالغ، یک جفت خرگوش دیگر تولید می کند.

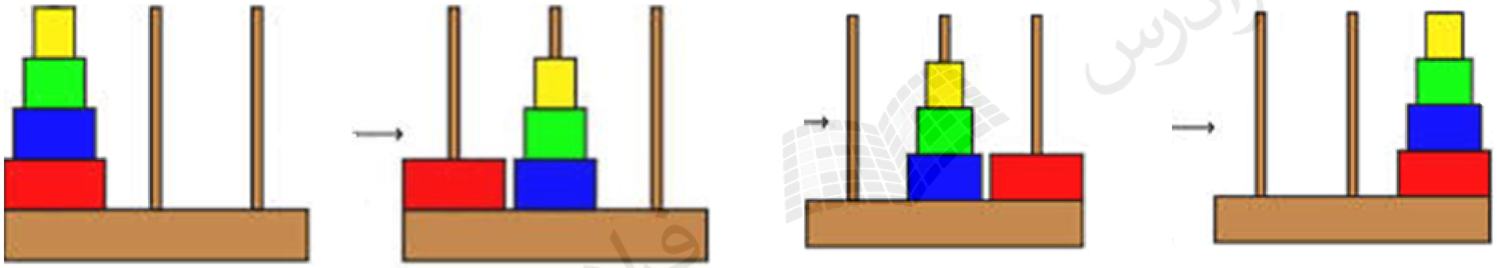
(ج) خرگوش ها هرگز نمی میرند.

رابطه بازگشتی برای تعداد خرگوش ها در شروع ماه n ام :



$$\begin{cases} f(n) = f(n-1) + f(n-2), n \geq 3 \\ f(1) = 1, f(2) = 1 \end{cases}$$

برج هانوی



$$a_n = a_{n-1} + 1 + a_{n-1}$$

$$a_n = 2a_{n-1} + 1$$

$$a_1 = 1$$

$$\Rightarrow a_n = 2^n - 1$$

روش های حل رابطه های بازگشتی

۱- تکرار با جایگذاری

۲- درخت بازگشت

۳- قضیه اصلی

۴- رابطه های بازگشتی همگن

مثال

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

$$T(1) = 1$$

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

مثال

مرتبه زمانی الگوریتم مقابل :

```
f(n)
{
  if(n>0)
  {
    f(n-1);
    print(n);
    f(n-1);
  }
}
```

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

$$T(n) \in \theta(2^n)$$

مثال

$$\begin{cases} T(n) = \dfrac{1}{n} + T(n-1) \\ T(1) = 1 \end{cases}$$

$$\begin{aligned} T(n) &= \frac{1}{n} + T(n-1) \\ &= \frac{1}{n} + \frac{1}{n-1} + T(n-2) \\ &= \frac{1}{n} + \frac{1}{n-1} + \frac{1}{n-2} + T(n-3) \\ &\;\;.... \\ &= \frac{1}{n} + \frac{1}{n-1} + \frac{1}{n-2} + \ldots + \frac{1}{2} + T(1) \\ &= \frac{1}{n} + \frac{1}{n-1} + \frac{1}{n-2} + \ldots + \frac{1}{2} + \frac{1}{1} \\ &\approx \ln n \end{aligned}$$

درخت بازگشت

روش درخت بازگشت(recursion tree)

به کمک این روش می توان رابطه های بازگشتی را حدس یا حل کرد.

در این روش نحوه جای گذاری یک عبارت بازگشتی و نیز مقدار ثابتی را که در هر سطح از آن عبارت به دست می آید نشان داده می شود. با جمع کردن مقادیر ثابت تمام سطوح، جواب بدست می آید.

درخت های بازگشت زمانی که رابطه بازگشتی، زمان اجرای یک الگوریتم تقسیم و حل را توصیف می کند مفید هستند.

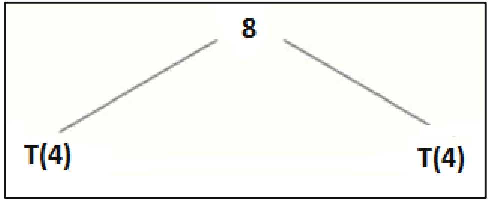
مثال

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

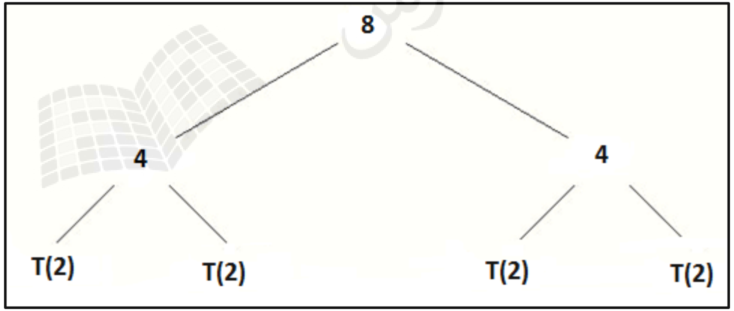
$T(1) = 1$

رسم درخت بازگشت $T(8)$:

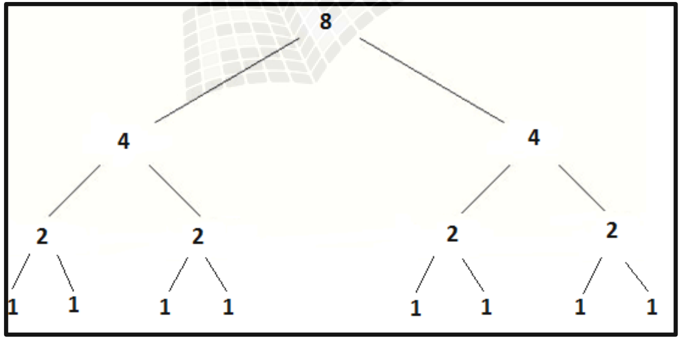
$T(8) = 2T(4) + 8$



$T(4) = 2T(2) + 4$

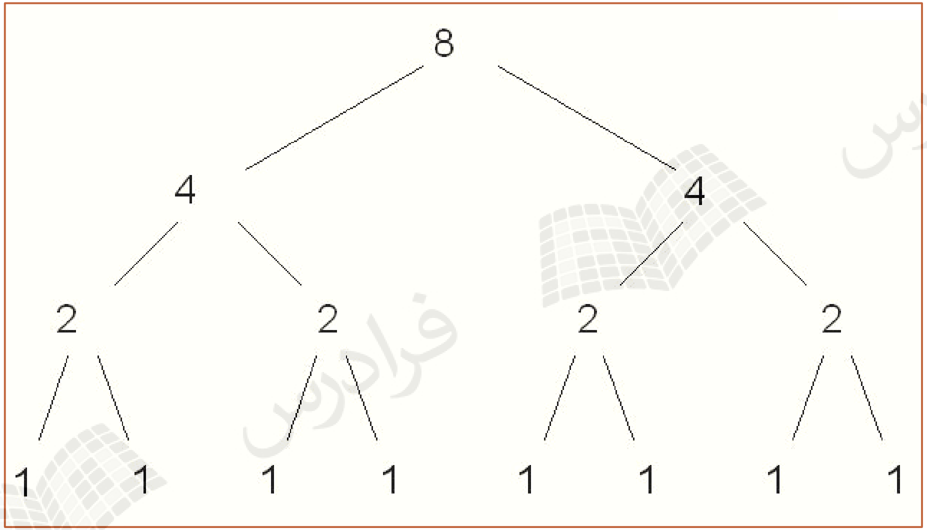


$T(2) = 2T(1) + 2$



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

$$T(1) = 1$$



هزینه:

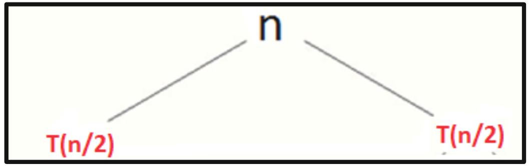
$$(8) + (4 + 4) + (2 + 2 + 2 + 2) + (1 + 1 + 1 + 1 + 1 + 1 + 1 + 1) = 4 \times 8 = 32$$

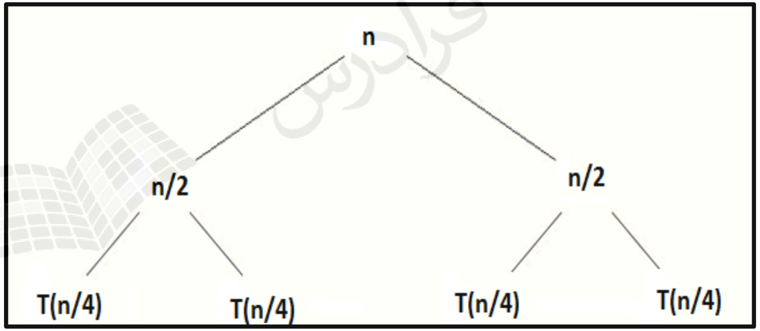
مثال

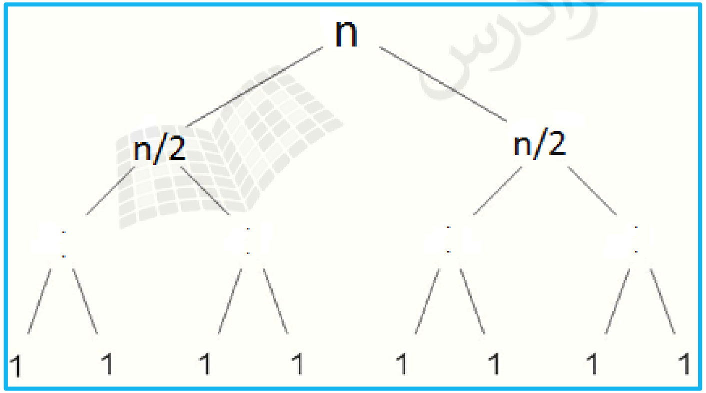
$$T(n) = 2T(\frac{n}{2}) + n$$

$$T(1) = 1$$

حل رابطه بازگشتی :



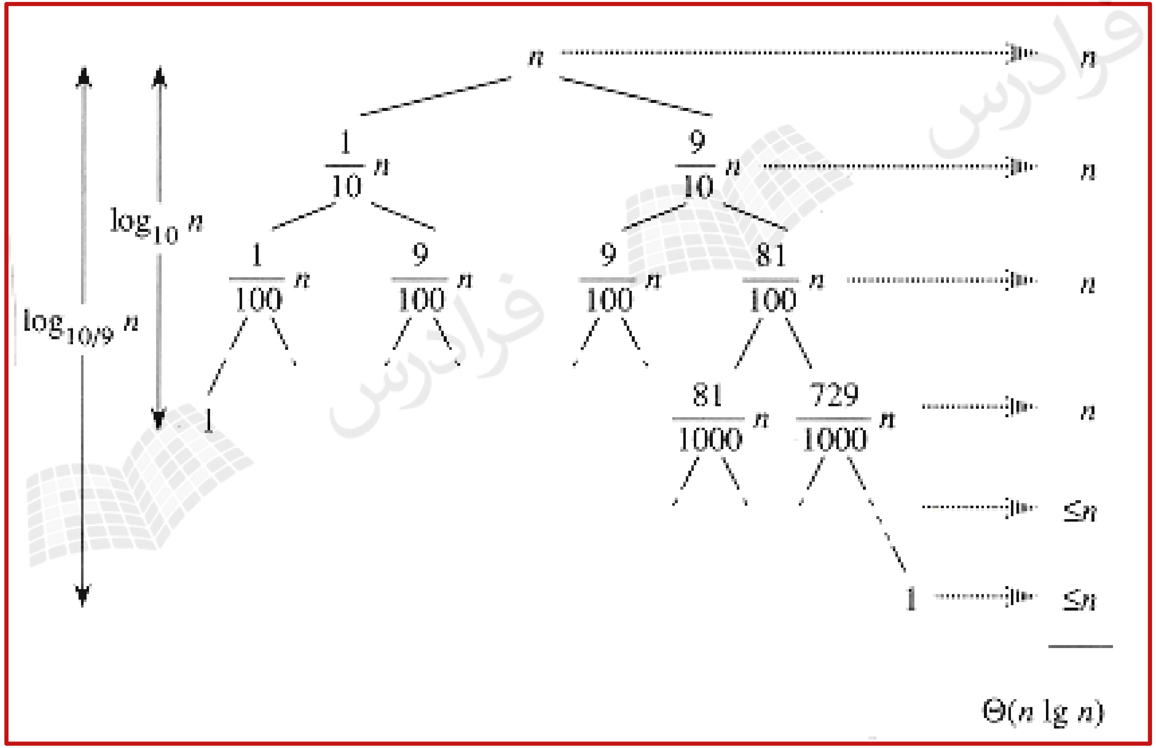




$$(\lg n + 1) \times n = n\lg n + n$$

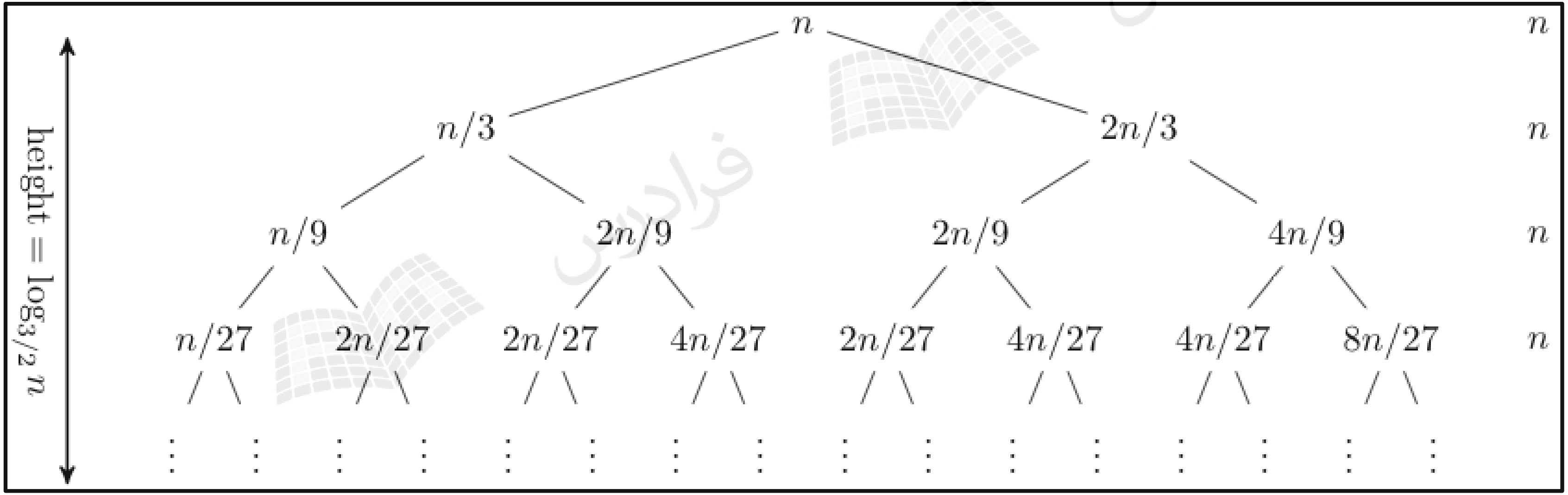
مثال

$$T(n) = T(\frac{n}{10}) + T(\frac{9n}{10}) + cn$$



مثال

$$T(n) = T(\frac{n}{3}) + T(\frac{2n}{3}) + n$$



فرمول

$$T(n) = T(\frac{n}{a}) + T(\frac{n}{b}) + cn \;\Rightarrow\; n\sum_{i=0}^{h}(\frac{1}{a}+\frac{1}{b})^i$$

ارتفاع درخت: حداکثر مقدار بین ارتفاع سمت چپ ($\log_a^n$) و ارتفاع سمت راست ($\log_b^n$)

$$T(n) = T(\frac{n}{10}) + T(\frac{9n}{10}) + n \;\Rightarrow\; T(n) = n\sum_{i=0}^{h}(\frac{1}{10}+\frac{9}{10})^i$$

$$= n\sum_{i=0}^{h}(1)^i = \theta(n\lg n)$$

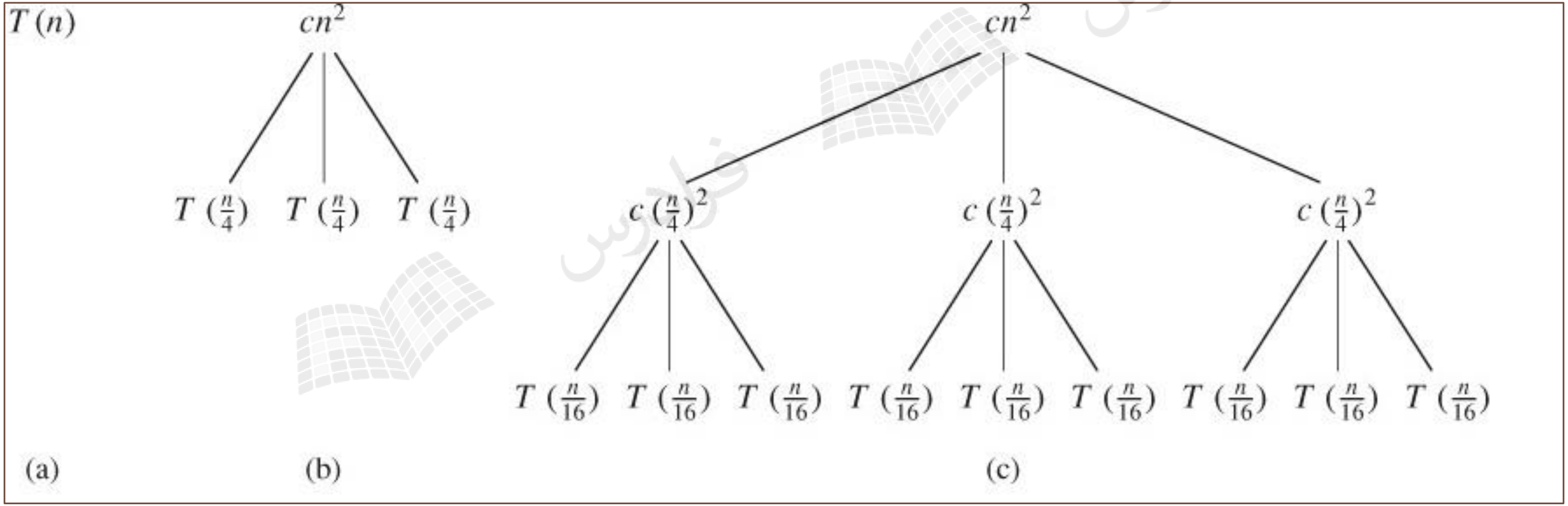
مثال

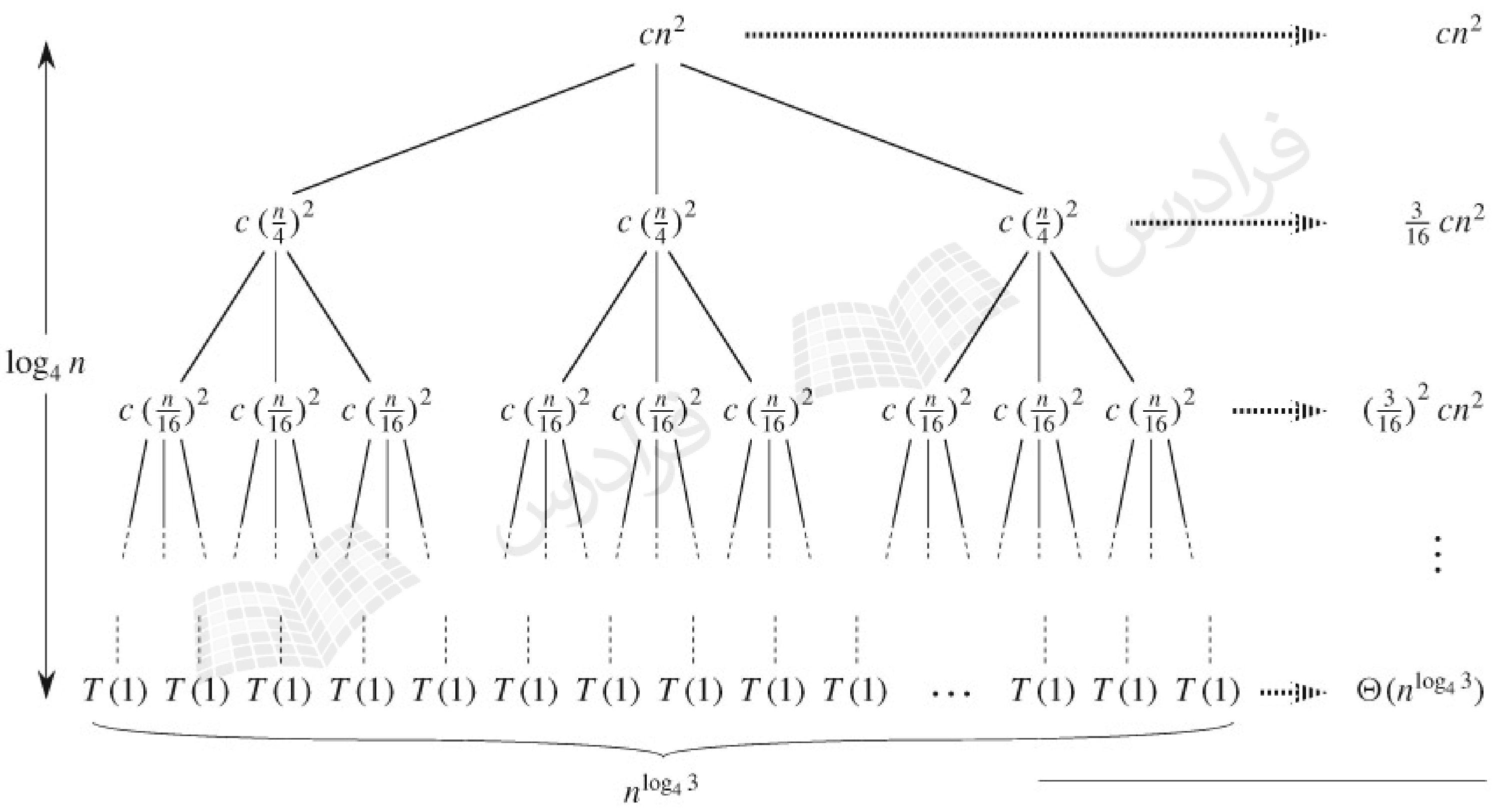
$$T(n) \le T\left(\frac{n}{5}\right) + T\left(\frac{7n}{10}\right) + n$$

$$T(n) \le n \sum_{i=0}^{\log_{10/7}^{n}} \left(\frac{1}{5} + \frac{7}{10}\right)^{i} = n \sum_{i=0}^{\log_{10/7}^{n}} \left(\frac{9}{10}\right)^{i}$$

مثال

$$T(n) = 3T(\frac{n}{4}) + cn^2$$





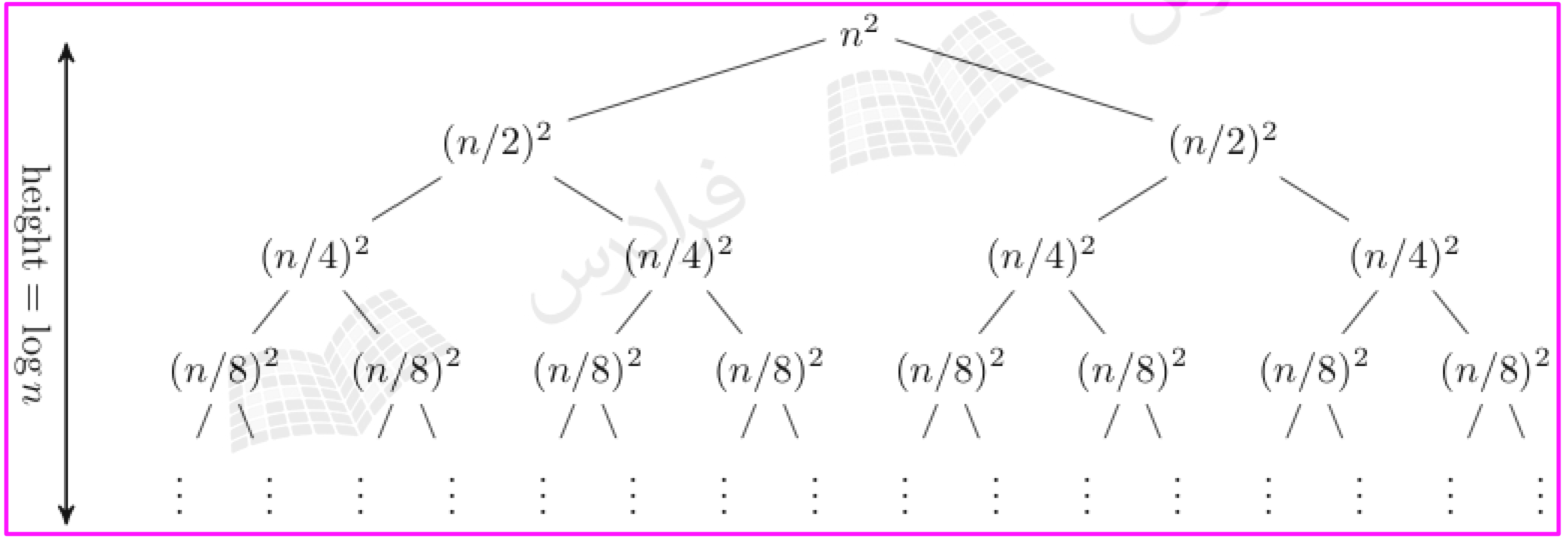
cn^2
cn^2
$c\left(\frac{n}{4}\right)^2$
$\frac{3}{16}cn^2$
$c\left(\frac{n}{16}\right)^2$
$\left(\frac{3}{16}\right)^2 cn^2$
$\log_4 n$
$T(1)$
$\Theta(n^{\log_4 3})$
$n^{\log_4 3}$

$$\begin{aligned}T(n) &= cn^2 + \frac{3}{16}cn^2 + \left(\frac{3}{16}\right)^2 cn^2 + \cdots + \left(\frac{3}{16}\right)^{\log_4 n-1} cn^2 + \Theta(n^{\log_4 3}) \\ &= \sum_{i=0}^{\log_4 n-1} \left(\frac{3}{16}\right)^i cn^2 + \Theta(n^{\log_4 3}) \\ &= \frac{(3/16)^{\log_4 n} - 1}{(3/16) - 1} cn^2 + \Theta(n^{\log_4 3}) .\end{aligned}$$

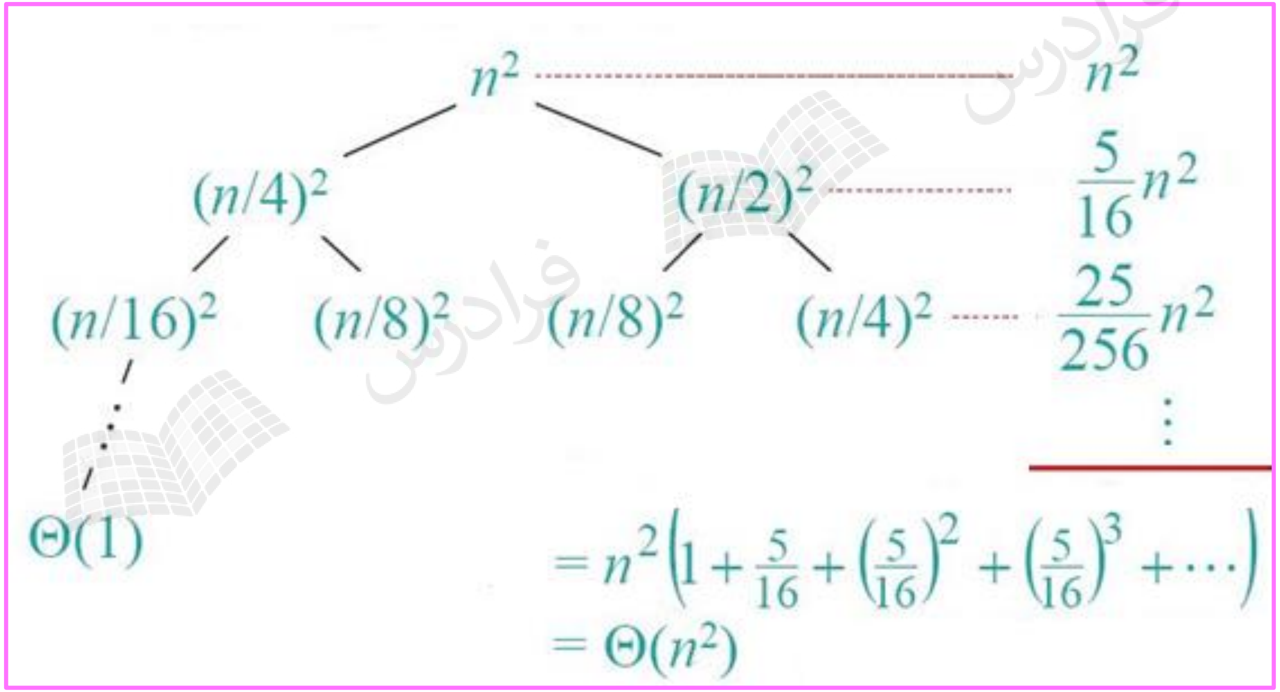
$$\begin{aligned}T(n) &= \sum_{i=0}^{\log_4 n-1} \left(\frac{3}{16}\right)^i cn^2 + \Theta(n^{\log_4 3}) \\ &< \sum_{i=0}^{\infty} \left(\frac{3}{16}\right)^i cn^2 + \Theta(n^{\log_4 3}) \\ &= \frac{1}{1-(3/16)} cn^2 + \Theta(n^{\log_4 3}) \\ &= \frac{16}{13} cn^2 + \Theta(n^{\log_4 3}) \\ &= O(n^2) .\end{aligned}$$

مثال

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



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

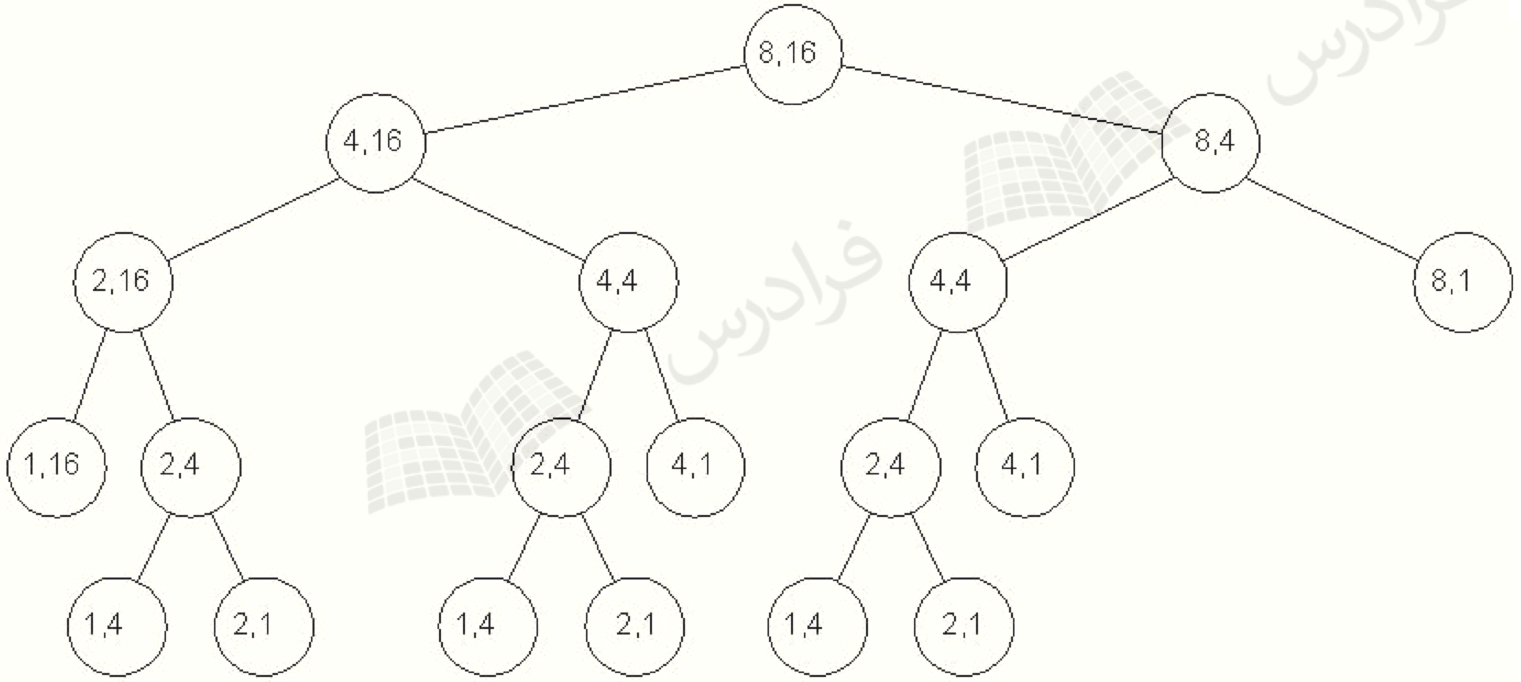


رابطه بازگشتی
دو متغیره

رابطه بازگشتی دو متغیره

$$T(n,k) = T(\frac{n}{2},k) + T(n,\frac{k}{4}) + kn$$

رسم درخت بازگشت $T(8,16)$:



ارتفاع:

$$\log_2^n + \log_4^k - 1$$

$$T(n,k)=T(\frac{n}{2},k)+T(n,\frac{k}{4})+kn$$

$$=nk+\left(\frac{n}{2}\times k+n\times\frac{k}{4}\right)+\left(\frac{n}{4}\times k+\frac{n}{2}\times\frac{k}{4}+\frac{n}{2}\times\frac{k}{4}+n\times\frac{k}{16}\right)+\dots$$

$$=nk+\frac{3}{4}nk+\frac{9}{16}nk+\dots$$

$$=nk\left(1+\frac{3}{4}+(\frac{3}{4})^2+\dots\right)\approx\theta(nk)$$

رابطه بازگشتی دو متغیره

$$T(n,k) = T\left(n_1, \left\lfloor \frac{k}{2} \right\rfloor\right) + T\left(n_2, \left\lfloor \frac{k}{2} \right\rfloor\right) + nk$$

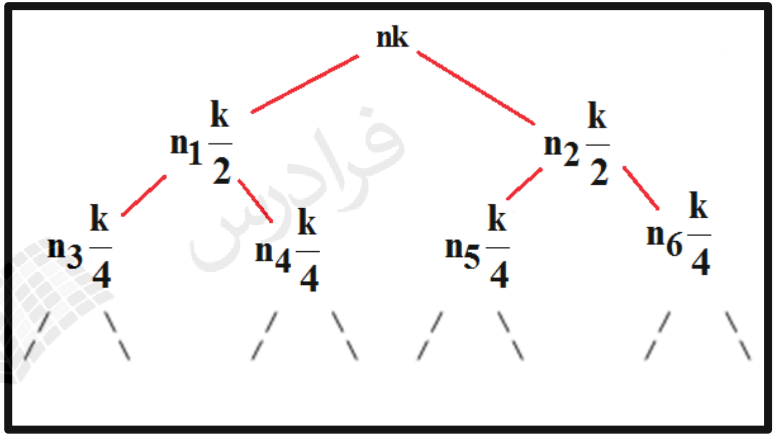
$$T(n,1) = T(1,k) = 1$$

$$nk + \left(n_1\frac{k}{2} + n_2\frac{k}{2}\right) + \left(n_3\frac{k}{4} + n_4\frac{k}{4} + n_5\frac{k}{4} + n_6\frac{k}{4}\right) + \ldots$$

$$= nk + (n_1 + n_2)\frac{k}{2} + (n_3 + n_4 + n_5 + n_6)\frac{k}{4} + ..$$

$$= nk + n\frac{k}{2} + (n_1 + n_2)\frac{k}{4} + \ldots$$

$$= nk + n\frac{k}{2} + n\frac{k}{4} + \ldots = nk\left(1 + \frac{1}{2} + \frac{1}{4} + \ldots\right) \approx nk$$



$(n_1 + n_2 = n)$

$(n_3 + n_4 = n_1, n_5 + n_6 = n_2)$

قضیه اصلی

قضیه اصلی

اگر داشته باشیم: $a \geq 1, b > 1$

$$T(n) = aT\left(\frac{n}{b}\right) + f(n)$$

$$T(n) = \begin{cases} \theta\left(n^{\log_b^a}\right) & f(n) < n^{\log_b^a} \\ \theta(f(n).\lg n) & f(n) = n^{\log_b^a} \\ \theta(f(n)) & f(n) > n^{\log_b^a} \end{cases}$$

مثال

$$T(n) = 4T(\frac{n}{2}) + \lg n$$

$$a = 4$$

$$b = 2$$

$$f(n) = \lg n$$

$$\lg n < n^{\log_2^4}$$

$$\theta(n^2)$$

مثال

$$T(n) = 3T(\frac{n}{4}) + n\lg n$$

$$n\lg n > n^{\log_4^3}$$

$$\theta(n\lg n)$$

مثال

$$T(n) = T(\frac{2n}{3}) + 1$$

$$1 = n^{\log_{3/2}^{1}}$$

$$\theta(\lg n)$$

نکته

در قضیه اصلی، اگر $\frac{f(n)}{n^{\log_b^a}} < n^{\varepsilon}$ باشد، یعنی $f(n)$ به صورت چند جمله ای از $n^{\log_b^a}$ بزرگتر نباشد ،

اگر $f(n)$ از مرتبه $n^{\log_b^a} . \lg^k n$ باشد، آنگاه مرتبه $T(n)$ برابر است با:

$$n^{\log_b^a} . \lg^{k+1} n$$

مثال

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

$$a = 2$$
$$b = 2$$
$$f(n) = n\lg n$$

$$\frac{n\lg n}{n} = \lg n < n^{\varepsilon}$$

$$T(n) = \theta(n\lg^2 n)$$

مثال

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

$$T(n) = \theta(n^0 (\lg n)^{2+1}) = \theta((\lg n)^3)$$

تغییر متغیر

مثال

مرتبه اجرایی رابطه بازگشتی $T(n) = T(\sqrt{n}) + 1$ مشخص کنید.

فرض: $n = 2^m$

$$T(2^m) = S(m) \Rightarrow T(2^{\frac{m}{2}}) = S(\frac{m}{2})$$

$$T(2^m) = T(2^{\frac{m}{2}}) + 1$$

$$S(m) = S(\frac{m}{2}) + 1 \Rightarrow S(m) = \theta(\lg m)$$

$$T(n) = \theta(\lg \lg n)$$

مثال

$$\mathbf{T(n) = 2T(\lfloor \sqrt{n} \rfloor) + lg\,n}$$

$$T(2^m) = 2T(2^{m/2}) + \lg 2^m$$

$$S(m) = 2S(\frac{m}{2}) + m$$

$$\theta(m \lg m)$$

$$\theta(\lg n . \lg \lg n)$$

$$T(n) = 4T(\sqrt{n}) + 1$$

$$n = 2^m$$

$$T(2^m) = 4T(2^{m/2}) + 1$$

$$S(m) = 4S(\frac{m}{2}) + 1$$

$$S(m) = \theta(m^{\log_2^4}) = \theta(m^2)$$

$$T(n) = \theta(\lg n)^2$$

رابطه های بازگشتی همگن

برای حل روابط بازگشتی همگن مرتبه دوم با ضرایب ثابت، ابتدا معادله مشخصه آن را پیدا می کنیم.

جواب بعد از حل این معادله با فرض داشتن:

الف– دو جواب مجزای r_1 , r_2 :

$$c_1 r_1^n + c_2 r_2^n$$

ب– یک ریشه حقیقی مضاعف r :

$$c_1 r^n + c_2 n r^n$$

مثال

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

$$T(0) = 2$$

$$T(1) = 7$$

$$T(n) - T(n-1) - 2T(n-2) = 0$$

$$r^2 - r - 2 = 0 \quad \Rightarrow r_1 = 2, r_2 = -1$$

جواب کلی:

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

$$T(0) = 2 \Rightarrow 2 = c_1 2^0 + c_2(-1)^0 \quad \Rightarrow c_1 + c_2 = 2$$

$$T(1) = 7 \Rightarrow 7 = c_1 2^1 + c_2(-1)^1 \quad \Rightarrow 2c_1 - c_2 = 7$$

$$\Rightarrow c_1 = 3, c_2 = -1$$

$$T(n) = 3 \times 2^n - (-1)^n$$

مثال

$$T(n+2) = 4T(n+1) - 4T(n)$$

$$T(0) = 1$$

$$T(1) = 3$$

$$T(n+2) - 4T(n+1) + 4T(n) = 0$$

$$r^2 - 4r + 4 = 0 \qquad \Rightarrow r_1 = 2, r_2 = 2$$

جواب کلی:

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

$$T(0) = 1 \Rightarrow c_1 \times 2^0 + c_2 \times 0 \times 2^0 = 1 \qquad \Rightarrow c_1 = 1$$

$$T(1) = 3 \Rightarrow c_1 \times 2^1 + c_2 \times 1 \times 2^1 = 3 \qquad \Rightarrow 2c_1 + 2c_2 = 3$$

$$c_1 = 1, c_2 = \frac{1}{2}$$

$$T(n) = 2^n + n2^{n-1}$$

مثال

$$T(n) = 3T(n-1) + 4T(n-2)$$

$$T(0) = 0, T(1) = 1$$

$$r^2 - 3r - 4 = 0$$

$$T(n) = c_1 4^n + c_2 (-1)^n$$

$$T(n) = \theta(4^n)$$

این اسلاید ها بر مبنای نکات مطرح شده در فرادرس

«آموزش طراحی الگوریتم»

تهیه شده است.

برای کسب اطلاعات بیشتر در مورد این آموزش به لینک زیر مراجعه نمایید

faradars.org/fvsft1092