Lec 11

Recurrence

Iterating a Recurrence

$Let's\ T(n) = rT(n-1) + a$ implies that:

$\forall i < n,\ T(n-i) = rT((n-i)-1)) + a$

Then, we have

$= rT(n-2) + a$

$$
\begin{aligned}
T(n) &= rT(n-1) + a \\
&= r(rT(n-2) + a) + a \qquad rT(n-3) + a \\
&= r^2T(n-2) + ra + a \\
&= r^2(rT(n-3) + a) + ra + a \qquad rT(n-4) + a \\
&= r^3T(n-3) + r^2a + ra + a \\
&= r^3(rT(n-4) + a) + r^2a + ra + a \\
&= r^4T(n-4) + r^3a + r^2a + ra + a.
\end{aligned}
$$

Guess $T(n) = r^nT(0) + a\sum_{i=0}^{n-1} r^i$

Theorem

If $T(n) = rT(n-1) + a,\ T(0) = b\ and\ r \neq 1,\ then:$

$$T(n) = r^nb + a\frac{1-r^n}{1-r}$$

First-Order Linear Recurrences

A recurrence of the form $T(n) = f(n)T(n-1) + g(n)$ is called a *first-order linear recurrence*.

◇ First Order because it only depends upon going back one step, i.e., $T(n-1)$

If it depends upon $T(n-2)$, it would be a second-order recurrence, e.g., $T(n) = T(n-1) + 2T(n-2)$.

Theorem

if $T(n) = \begin{cases} rT(n-1) + g(n) & if\ n > 0 \\ a & if\ n = 0 \end{cases}$ we can find :

$$T(n) = r^n a + \sum_{i=1}^{n} r^{n-i} g(i)$$

$$or$$

$$T(n) = r^n a + \sum_{i=0}^{n-1} r^{i} g(n-i)$$

$$(i = n - j \qquad j = n - i)$$

$$or$$

$$T(n) = r^n a + \sum_{j=1}^{n} r^{n-j} g(j)$$

等比数列求和:

通项式：$a_n = a_1 * q^{n-1}$

和：$S_n = \frac{a_1*(1-q^n)}{1-q}$

Theorem

For all real number x $\neq$ 1,

$$\sum_{i=1}^{n} ix^i = \frac{nx^{n+2} - (n+1)x^{n+1} + x}{(1-x)^2}$$

proof :

$$\sum_{i=1}^{n} ix^i$$

① $$\left(\sum_{i=1}^{n} x^i\right)' = \left(\frac{x(1-x^n)}{1-x}\right)'$$

$$\sum_{i=1}^{n} ix^{i-1} = \frac{(x-x^{n+1})'(1-x) - (x-x^{n+1})(1-x)'}{(1-x)^2} = \frac{nx^{n+1} - (n+1)x^n + 1}{(1-x)^2}$$

multiply both sides by x

$$\sum_{i=1}^{n} ix^i = \frac{nx^{n+2} - (n+1)x^{n+1} + x}{(1-x)^2}$$

$$\sum_{i=1}^{n} ix^i = S$$

② $$\sum_{i=1}^{n} ix^{i+1} = xS$$

$$(1-x)S = \sum_{i=1}^{n} ix^i - \sum_{i=1}^{n} ix^{i+1}$$

$$x + 2x^2 + 3x^3 + \cdots + nx^n$$

$$- \qquad x^2 + 2x^3 + \cdots + (n-1)x^n + nx^{n+1}$$

$$= x + \cancel{2}x^2 + x^3 + \cdots + x^n - nx^{n+1}$$

$$= \frac{x(1-x^n)}{1-x} - nx^{n+1}$$

Divide and conquer algorithms

binary Search

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

Example

- $$(*) \qquad T(n) = \begin{cases} C_1 & \text{if } n = 1 \\ T(\lceil n/2 \rceil) + C_2 & \text{if } n \geq 2 \end{cases}$$

For simplicity, we will (usually) assume that n is a power of 2 (or sometimes 3 or 4) and also often that constants such as C_1, C_2 are 1. This will let us replace a recurrence such as (*) by one such as (**).

$$\begin{aligned} T(n) &= 2^i T(\tfrac{n}{2^i}) + in \qquad\qquad i = log_2 n \\ &= 2^{log_2 n} T(1) + log_2 n * n \\ &= nT(1) + nlog_2 n \end{aligned}$$

$$(**) \qquad T(n) = \begin{cases} 1 & \text{if } n = 1 \\ T(n/2) + 1 & \text{if } n \geq 2 \end{cases}$$

In practice, the solution of (*) will be very close to that of (**) (this can be proved mathematically). Hence, we can restrict attention to (**).

$$\begin{aligned} T(n) &= T\left(\tfrac{n}{2}\right) + 1 && = \left(T\left(\tfrac{n}{2^2}\right) + 1\right) + 1 \\ &= T\left(\tfrac{n}{2^2}\right) + 2 && = \left(T\left(\tfrac{n}{2^3}\right) + 1\right) + 2 \\ &= T\left(\tfrac{n}{2^3}\right) + 3 \\ &\ \ \vdots \qquad \vdots \\ &= T\left(\tfrac{n}{2^i}\right) + i \end{aligned}$$

$$\begin{aligned} T(n) &= T(\tfrac{n}{2^i}) + i \qquad\qquad i = log_2 n \\ &= T(n/2^{log_2 n}) + log_2 n \\ &= T(1) + log_2 n \\ &= 1 + log_2 n \end{aligned}$$

$$(*) \qquad T(n) = \begin{cases} 1 & \text{if } n = 1 \\ T(n/2) + n & \text{if } n \geq 2 \end{cases}$$

$$\begin{aligned} T(n) &= T\left(\tfrac{n}{2}\right) + n \\ &= T\left(\tfrac{n}{2^2}\right) + \tfrac{n}{2} + n \\ &= T\left(\tfrac{n}{2^3}\right) + \tfrac{n}{2^2} + \tfrac{n}{2} + n \\ &\quad\vdots \qquad \vdots \\ &= T\left(\tfrac{n}{2^i}\right) + \tfrac{n}{2^{i-1}} + \cdots + \tfrac{n}{2^2} + \tfrac{n}{2} + n \\ &\quad\vdots \qquad \vdots \\ &= T\left(\tfrac{n}{2^{\log_2 n}}\right) + \tfrac{n}{2^{\log_2 n - 1}} + \cdots + \tfrac{n}{2^2} + \tfrac{n}{2} + n \\ &= 1 + 2 + 2^2 + \cdots + \tfrac{n}{2^2} + \tfrac{n}{2} + n \quad = \Theta(n) \end{aligned}$$

30

$$(*) \qquad T(n) = \begin{cases} 1 & \text{if } n < 3 \\ 3T(n/3) + n & \text{if } n \geq 3 \end{cases}$$

Assume n is a power of 3

$$\begin{aligned} T(n) &= 3T(n/3) + n \\ &= 3\left(3T(n/3^2) + n/3\right) + n \\ &= 3^2 T(n/3^2) + 2n \\ &= 3^2\left(3T(n/3^3) + n/3^2\right) + 2n \\ &= 3^3 T(n/3^3) + 3n \\ &\quad\vdots \\ &= 3^i T(n/3^i) + in \\ &\quad\vdots \\ &= 3^{\log_3 n} T\left(n/3^{\log_3 n}\right) + n\log_3 n = n + n\log_3 n \end{aligned}$$

31

Example 5

$$(*) \qquad T(n) = \begin{cases} 1 & \text{if } n = 1 \\ 4T(n/2) + n & \text{if } n \geq 2 \end{cases}$$

$T(n) = 4\underline{T(n/2)} + n \qquad T(n/2) = 4T(n/2^2) + n/2$

$= 4(4T(n/2^2) + n/2) + n$

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

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

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

$\vdots$

$= 4^i T(n/2^i) + \frac{4^{i-1}}{2^{i-1}}n + \cdots + \frac{4}{2}n + n$

$\vdots$

$= \underline{4^{\log_2 n} T(n/2^{\log_2 n})} + \frac{4^{\log_2 n - 1}}{2^{\log_2 n - 1}}n + \cdots + \frac{4}{2}n + n$

$= n^2 + n(1 + 2 + \cdots + 2^{\log_2 n - 1})$

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

$= 2n^2 - n$

32

Three Different Behaviors

- **Theorem** Suppose that we have a recurrence of the form
$$T(n) = aT(n/2) + n,$$
where a is a positive integer and $T(1)$ is nonnegative. Then we have the following big Θ bounds on the solution:

  1. If $\underline{a < 2}$, then $T(n) = \Theta(n)$.
  2. If $\underline{a = 2}$, then $T(n) = \Theta(n \log n)$. Merge Sort
  3. If $\underline{a > 2}$, then $T(n) = \Theta(n^{\log_2 a})$

Total work

- n times the largest term in the geometric series is

$$n\left(\frac{a}{2}\right)^{\log_2 n-1} = \frac{2}{a}\cdot\frac{n\cdot a^{\log_2 n}}{2^{\log_2 n}} = \frac{2}{a}\cdot\frac{n\cdot a^{\log_2 n}}{n} = \frac{2}{a}\cdot a^{\log_2 n}$$

$$a^{\log_2 n} = (2^{\log_2 a})^{\log_2 n} = (2^{\log_2 n})^{\log_2 a} = n^{\log_2 a}$$

Notice that

$$a^{\log_2 n} = \left(2^{\log_2 a}\right)^{\log_2 n} = \left(2^{\log_2 n}\right)^{\log_2 a} = n^{\log_2 a}$$

So the total work is

$$a^{\log_2 n}T(1) + n\sum_{i=0}^{\log_2 n-1}\left(\frac{a}{2}\right)^i$$

$$\Theta\left(n^{\log_2 a}\right) \qquad \Theta\left(n^{\log_2 a}\right)$$

The master Theorem

Theorem Suppose that we have a recurrence of the form

$$T(n) = aT(n/b) + cn^d,$$

where a is a positive integer, $b \geq 1$, c, d are real numbers with c positive and d nonnegative, and $T(1)$ is nonnegative. Then we have the following big Θ bounds on the solution:

1. If $a < b^d$, then $T(n) = \Theta(n^d)$.
2. If $a = b^d$, then $T(n) = \Theta(n^d \log n)$.
3. If $a > b^d$, then $T(n) = \Theta(n^{\log_b a})$