Advanced Counting Techniques

CS1200, CSE IIT Madras

Meghana Nasre

April 9, 2020

Advanced Counting Techniques

- Principle of Inclusion-Exclusion ✓
- Recurrences and its applications ✓
- Solving Recurrences

Recurrences

Recurrences occur many times especially in analysis of algorithms.

Our goal: To obtain a closed form solution to the recurrence.

Why closed form?

- We may want the 1000-th term which depends on the 999-th term. Computing that with the recursive formulation is tedious.
- The recurrence may denote the running time of your algorithm. We want an estimate on running time.

Today's class: (Some) Techniques to solve recurrences.

There is no single recipe to solve all recurrences. However, we will show techniques that apply to a wide variety.

Some familiar recurrences

Fibonacci Sequence Recurrence

$$f(n) = \begin{cases} n & \text{if } n = 0 \text{ or } n = 1 \\ f(n-1) + f(n-2) & \textit{otherwise} \end{cases}$$

Towers of Hanoi Recurrence

$$T(n) = \begin{cases} 1 & \text{if } n = 1 \\ 2T(n-1) + 1 & \textit{otherwise} \end{cases}$$

Binary Search Recurrence

assume $n = 2^k$ for $k \geq 0$

$$T(n) = \begin{cases} 1 & \text{if } n = 1 \\ T\left(\frac{n}{2}\right) + 1 & \textit{otherwise} \end{cases}$$

Repeated Substitution Method : Example 1

Binary Search Recurrence assume $n = 2^k$ for $k \geq 0$

$$
\begin{aligned}
T(n) &= 1 && \text{if } n = 1 \\
&= T\left(\frac{n}{2}\right) + 1 && \textit{otherwise}
\end{aligned}
$$

$$
\begin{aligned}
T(n) &= T\left(\frac{n}{2}\right) + 1 \\
&= T\left(\frac{n}{4}\right) + 1 + 1 &&= T\left(\frac{n}{2^2}\right) + 2 \cdot 1 \\
&= T\left(\frac{n}{8}\right) + 1 + 1 + 1 &&= T\left(\frac{n}{2^3}\right) + 3 \cdot 1 \\
&\vdots \\
&= T\left(\frac{n}{2^k}\right) + k \cdot 1 && \text{observing a pattern}
\end{aligned}
$$

At this step we can guess that $\boxed{T(n) = \log n + 1}$. Recall $n = 2^k$ for $k \geq 0$.

How do we confirm the guess? Ex: Use induction on n to prove the guess.

Repeated Substitution Method : Example 2

Towers of Hanoi Recurrence

$$T(n) = \begin{cases} 1 & \text{if } n = 1 \\ 2T(n-1)+1 & \textit{otherwise} \end{cases}$$

$$\begin{aligned} T(n) &= 2T(n-1)+1 \\ &= 2(2T(n-2)+1)+1 \\ &= 2^2T(n-2)+2+1 \\ &= 2^2(2T(n-3)+1)+2+1 \\ &= 2^3T(n-3)+2^2+2+1 \\ &\vdots \\ &= 2^kT(n-k)+\sum_{i=0}^{k-1}2^i \qquad \text{observing a pattern} \end{aligned}$$

At this step we can guess that $\boxed{T(n) = 2^n - 1}$.

How do we confirm the guess? Ex: Use induction on n to prove the guess.

Repeated Substitution Method : Example 3

Another recurrence

$$T(n) = \begin{cases} 0 & \text{if } n = 0 \\ 3T(n-1) + 2n & \textit{otherwise} \end{cases}$$

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

work out and observe a pattern

$$\vdots$$

$$\begin{aligned} &= 3^k T(n-k) + 2\sum_{i=0}^{k-1} 3^i (n-i) \\ &= 3^n T(0) + 2\sum_{i=0}^{n-1} 3^i (n-i) \\ &= 2n\left(\sum_{i=0}^{n-1} 3^i\right) - 2\left(\sum_{i=0}^{n-1} 3^i i\right) \end{aligned}$$

we need to solve $\sum_{i=0}^{n-1} 3^i i$

Repeated Substitution Method : Example 3

Another recurrence

$$T(n) = \begin{cases} 0 & \text{if } n = 0 \\ 3T(n-1) + 2n & \textit{otherwise} \end{cases}$$

$$\begin{aligned} T(n) &= 3T(n-1) + 2n \\ &= 2n\left(\sum_{i=0}^{n-1} 3^i\right) - 2\left(\sum_{i=0}^{n-1} 3^i i\right) && \text{we need to solve } \textstyle\sum_{i=0}^{n-1} 3^i i \\ &= \frac{2n \cdot 3^n + 3^{n+1} - 4n - 3}{4} && \text{needs to be proved by induction} \end{aligned}$$

A sub-problem to solve first.

$$\left(\sum_{i=0}^{n-1} 3^i i\right) = \frac{3^n(2n-3) + 3}{4} \qquad \text{this needs a derivation and proof!}$$

In general: good to know sum of commonly occurring sums.

Repeated Substitution Method : Learnings

- An elementary method to solve recurrences.
  elementary does not mean simple, but a something that does not need background
- Need to observe a pattern.
- Oversimplification may make us miss the pattern.
- Creativity and experience with summation of series help.
- However, the pattern has to be observed for each recurrence and there is no generic rule. Are there some recurrences that can be solved by a formula?

Ex: Solve by repeated substitution

$$
\begin{aligned}
T(n) &= 2 && \text{if } n = 0 \\
&= 2\sqrt{T(n-1)} && \textit{otherwise}
\end{aligned}
$$

$$
\begin{aligned}
T(n) &= 12 && \text{if } n = 0 \\
&= 20 && \text{if } n = 1 \\
&= 2T(n-1) - T(n-2) && \textit{otherwise}
\end{aligned}
$$

Linear recurrences

Linear Homogeneous Recurrences with constant coefficients

$$a_n = c_1 a_{n-1} + c_2 a_{n-2} + \ldots + c_k a_{n-k}$$

$1 \leq i \leq k$, c_i is a real number and $c_k \neq 0$

- Linear because a_{n-1}, a_{n-2} ... appear in separate terms and to the first power.
- Homogeneous because degree of every term is the same. There is no constant term.
- Constant coefficients because $c_1, c_2 \ldots$ are reals which do not depend on n.

Examples:

- $T(n) = 2T(n-1)$ and $T(0) = 1$.
- $T(n) = T(n-1) + T(n-2)$ and $T(0) = 0, T(1) = 1$.

Non Examples:

- $T(n) = nT(n-1)$ and $T(0) = 1$ does not have constant coefficients
- $T(n) = T(n-1) \cdot T(n-2)$ and $T(0) = 0, T(1) = 1$. not linear
- $T(n) = n \cdot T(n-1)$ and $T(1) = 1$. does not have constant coefficients

Linear Homogeneous Recurrences with constant coefficients

$$a_n = c_1 a_{n-1} + c_2 a_{n-2} + \ldots + c_k a_{n-k}$$

$1 \leq i \leq k$, c_i is a real number and $c_k \neq 0$

Our goal: To derive a systematic way of solving such recurrences.
Upcoming next week

- A technique to solve linear homogeneous equations.
- A technique to solve linear non-homogeneous equations.
- References: Section 8.2[KR]