RECURRENCE RELATION AND PARTITIONS

Maruti Srinivas
07CS1030

Teacher Prof. Niloy Ganguly
Department of Computer Science and Engineering
IIT Kharagpur

19 August 2008

Contents

Abstract

Recurrence relations arise naturally in many counting problems and in analyzing the programming problems.One such example is analysis of the time complexity of an algorithm. Very often the time is obtained as a recurrence relation .In this lecture we discuss the definition and also some examples of how these relations basically work. Then the methods to solve the recursive equations are discussed. Partitions are discussed as part of Relations and Digraphs. Here also the definition and the concepts of partitioning a set into its equivalence classes is given. The basics of relations and partitions has been presented.

1 Recurrence Relations

1.1 Definition

Recursive formulae are commanly used to represent sequences. Its just like a compact notation to represent a sequence. In these equations previous terms in the sequence are referred to define the next term. These are said to be recursive in nature. Every recursive formula has a starting place. Also some initial conditions must be given for the sequence. Here's a simple example :

$$T_n = T_{n-1} + 2 \times T_{n-2} \text{ and } T_0 = 2, T_1 = 3$$

Another very common example is the *Fibonacci Sequence*.
Many times we encounter questions where the recursive relation has to be determined. One such example is given here:

Solve: An annuity of Rs.10000 earns 8% compounded monthly. Each month Rs.250 is withdrawn from here.Write a recurrence relation for the monthly balance at the end of n months
Solution: Let T_{n-1} balance at the end of last month, i.e. n-1 th month. Since interest is compounded monthly the interest obtained for the nth month will be 8% of T_{n-1}. Thus interest is 0.08 $\times \mathrm{T}_{n-1}$. And since the amount of Rs.250 is withdrawn from the account the total balance is $T_{n-1} + 0.08T_{n-1} - 250$. Thus we can write the recursive relation as $T_n = T_{n-1} + 0.08T_{n-1} - 250$ with initial conditions $T_0 = 10000$.

1.2 Solving Recursions

Solving recursive equations is a common task encountered in many of programming problems and mathematics. Solving the recursive relation involves only finding the explicit formula for the nth term.There are mainly two methods to solve for the explicit formula of the nth term.

1.2.1 Backtracking

The technique for solving using backtracking is shown here. From the nth term equation given which consists of lower terms in the sequence the explicit formula is obtained. The lower terms are substituted in much lower terms and finally all terms on right are expressed in terms of n. Here's an example:
Solve: Find explicit formula for the sequence defined by $c_n = 2c_{n-1} + 1, c_1 = 7$.
Solution: The definition of the previous term is substituted in the formula.

$$\begin{aligned} c_n &= 2c_{n-1} + 1 \\ &= 2(2c_{n-2} + 1) + 1 \\ &= 2[2(2c_{n-3} + 1) + 1] + 1 \\ &= 2^3 c_{n-3} + 4 + 2 + 1 \\ &= 2^3 c_{n-3} + 2^2 + 2^1 + 1 \end{aligned}$$

Slowly a regular pattern is coming up.this backtracking ends at

$$\begin{aligned} c_n &= 2^{n-1} c_{n-(n-1)} + 2^{n-2} + 2^{n-3} + ... + 2^2 + 2^1 + 1 \\ &= 2^{n-1} c_1 + 2^{n-1} - 1 \\ &= 7.2^{n-1} + 2^{n-1} - 1 \\ &= 8.2^{n-1} - 1 \\ &= 2^{n+2} - 1 \end{aligned}$$

Backtracking can sometimes be cumbersome and lead to equations which are very hectic to handle. Next a very easy method of solving recursion relation.

1.2.2 Linear Homogeneous Relations

This a more general technique for solving a recurrence relation. A recurrence relation is a *linear homogeneous relation of degree k* if it is of the form

$$a_n = r_1 a_{n-1} + r_2 a_{n-2} + ... + r_k a_{n-k} \text{ with } r_i\text{'s constants}$$

Examples:
- c$_n = c_{n-1}$ is a linear homogeneous recurrence relation of degree 1.
- a$_n = a_{n-1} + 3$ is not a linear homogeneous recurrence relation.
- t$_n = t_{n-1}^2 + t_{n-2}$ is not a linear homogeneous recurrence relation.

Each linear homogeneous recurrence relation of degree k is associated with a polynomial of degree k known as its *characteristic equation*. Its given by :

$$x^k = r_1 x^{n-1} + r_2 x^{n-2} + ... + r_k$$

The roots of the characteristic equation are very importaant in expressing the explicit formula of the sequence. For a homogeneous equation of degree 2 the characteristic equation is $x^2 - r_1 x - r_2 = 0$

Theorem:
(a) If the characteristic equation $x^2 - r_1 x - r_2 = 0$ of a recurrence relation $a_n = r_1 a_{n-1} + r_2 a_{n-2}$ has two distinct roots , s_1 and s_2 , then $a_n = us_1^n + vs_2^n$,where u and v depend on initial conditions, is the explicit formula for the sequence.
(b) If the characteristic equation $x^2 - r_1 x - r_2 = 0$ has a single root s , then the explicit formula is $a_n = us^n + vns^n$,where u and v depend on initial conditions.
Proof:
Suppose that s_1 and s_2 are the roots of $x^2 - r_1 x - r_2 = 0$, so $s_1^2 - r_1 s_1 - r_2 = 0$, $s_2^2 - r_1 s_2 - r_2 = 0$ and $a_n = us_1^n + vs_2^n$, for $n \geq 1$. We show that the definition of a$_n definesthesamesequenceas$ a$_n$ $= r_1 a_{n-1} + r_2 a_{n-2}$. First we note that u and v are chosen so that $a_1 = us_1 + vs_2$ and $a_2 = us_1^2 + vs_2^2$ so the initial conditions are satisfied. Then

$$\begin{aligned}
a_n &= us_1^n + vs_2^n \\
&= us_1^{n-2} s_1^2 + vs_2^{n-2} s_2^2 \\
&= us_1^{n-2}(r_1 s_1 + r_2) + vs_2^{n-2}(r_1 s_2 + r_2) \\
&= r_1 us_1^{n-1} + r_2 us_2^{n-2} + r_2 vs_2^{n-2} \\
&= r_1(us_1^{n-1} + vs_2^{n-1}) + r_2(us_1^{n-2} + vs_2^{n-2}) \\
&= r_1 a_{n-1} + r_2 an - 2
\end{aligned}$$

Solve: Solve the recurrence relation $d_n = 2d_{n-1} - d_{n-2}$ with the initial conditions d_1 =1.5 and d_2 =3.
Solution: The corresponding characteristic equation is $x^2 - 2x + 1 = 0$. This equation has one root ,1. Thus using the theorem above and initial conditions we get $d_1 = 1.5 = u + v(1)$ and $d_2 = 3 = uv(2)$. Solving for u and v we get $u = 0$ and $v = 1.5$. Thus $d_n = 1.5n$ is the explicit formula.

2 Relations

2.1 Cartesian Products

If A and B are two nonempty sets , the cartesian product $A \times B$ is defined as the set of all ordered pairs (a, b) with $a \epsilon A$ and $b \epsilon B$.Thus

$$\text{A } \times B = \{(a, b) | a \epsilon A \text{ and } b \epsilon B\}.$$

Example: Let $A = \{1, 2, 3\}$ and $B = \{r, s\}$ Then the product

$$A \times B = \{\text{(1,r),(1,s),(2,r),(2,s),(3,r),(3,s)}\}.$$

2.2 Partitions

A **partition** of a nonempty set A is a collection $\wp$ of nonempty subsets of A such that

1.Each element in A belongs to one of the sets in $\wp$.

2.If A_1 and A_2 are distinct elements of $\wp$ then $A_1 \cap A_2 = \phi$.

The sets in $\wp$ are the **blocks** of the partition .

Example: Let $A = \{a, b, c, d, e, f, g, h\}$. Consider the following subsets of A :

$$A_1 = \{a, b, c, d\}, A_2 = \{a, c, e, f, g, h\}, A_3 = \{a, c, e, g\}, A_4 = \{b, d\} \text{ and } A_5 = \{f, h\}.$$

Then $\{A_1, A_2\}$ is not a partition since $A_1 \cap A_2 \neq \phi$. The collection $\wp = \{A_3, A_4, A_5\}$ is a partition of A.Similarly we can have many partition sets like this.

2.3 Bell Number

The nth **Bell number** is the number of partitions of a set with n members, or equivalently, the number of equivalence relations on it.

In general, B_n is the number of partitions of a set of size n. For example, $B_3 = 5$ because the 3-element set {a, b, c} can be partitioned in 5 distinct ways:
{ {a}, {b}, {c} }
{ {a}, {b, c} }
{ {b}, {a, c} }
{ {c}, {a, b} }
{ {a, b, c} }.

Bell Number follows the following recursive formula

$$B_{n+1} = \sum_{k=0}^{n} \left(\begin{array}{c} n \\ k \end{array} \right) B_k$$

****