Advanced Discrete Mathematics

Subhash G. Deo

This reference book can be useful for

BBA, MBA, B.Com, BMS, M.Com, BCA, MCA

and many more courses for Various Universities

NEERAJ PUBLICATIONS

www.neerajbooks.com

Published by:

NEERAJ PUBLICATIONS

(Publishers of Educational Books)

Sales Office : 1507, 1st Floor,
Nai Sarak, Delhi-110 006
E-mail: info@neerajbooks.com
Website: www.neerajbooks.com

Typesetting by: Competent Computers

Terms & Conditions for Buying E-Book

- The User must Read & Accept the Terms and Conditions (T&C) carefully before clicking on the accept option for Buying the Online Soft Copy of E-books. Under this Particular Facility you may buy only the Online Soft Copy of E-books, no Hard Copy or Printed Copy shall be provided under this facility.
- These E-Books are valid for 365 days online reading only (From the Date of Purchase) and no kind of Downloading, Printing, Copying, etc. are allowed in this facility as these products are just for Online Reading in your Mobile / Tablet / Computers.
- All the online soft copy E-books given in this website shall contain a diffused watermark on nearly every page to protect the material from being pirated / copy / misused, etc.
- This is a Chargeable Facility / Provision to Buy the Online Soft Copy of E-books available online through our Website Which a Subscriber / Buyer may Read Online on his or her Mobile / Tablet / Computer. The E-books content and their answer given in these Soft Copy provides you just the approximate pattern of the actual Answer. However, the actual Content / Study Material / Assignments / Question Papers might somewhat vary in its contents, distribution of marks and their level of difficulty.
- These E-Books are prepared by the author for the help, guidance and reference of the student to get an idea of how he/she can study easily in a short time duration. Content matter & Sample answers given in this E-Book may be Seen as the Guide/Reference Material only. Neither the publisher nor the author or seller will be responsible for any damage or loss due to any mistake, error or discrepancy as we do not claim the Accuracy of these solution / Answers. Any Omission or Error is highly regretted though every care has been taken while preparing these E-Books. Any mistake, error or discrepancy noted may be brought to the publishers notice which shall be taken care of in the next edition. Please consult your Teacher/Tutor or refer to the prescribed & recommended study material of the university / board / institute / Govt. of India Publication or notification if you have any doubts or confusions before you appear in the exam or Prepare your Assignments before submitting to the University/Board/Institute.
- Publisher / Study Badshah / shall remain the custodian of the Contents right / Copy Right of the Content of these reference E-books given / being offered at the website www.studybadshah.com.
- The User agrees Not to reproduce, duplicate, copy, sell, resell or exploit for any commercial purposes, any portion of these Services / Facilities, use of the Service / Facility, or access to the Service / Facility.
- The Price of these E-books may be Revised / Changed without any Prior Notice.
- The time duration of providing this online reading facility of 365 days may be alter or change by studybadshah.com without any Prior Notice.
- The Right to accept the order or reject the order of any E-books made by any customer is reserved with www.studybadshah.com only.
- All material prewritten or custom written is intended for the sole purpose of research and exemplary purposes only. We encourage you to use our material as a research and study aid only. Plagiarism is a crime, and we condone such behaviour. Please use our material responsibly.
- In any Dispute What so ever Maximum Anyone can Claim is the Cost of a particular E-book which he had paid to Study Badshah company / website.
- If In case any Reader/Student has paid for any E-Book and is unable to Access the same at our Website for Online Reading Due to any Technical Error/ Web Admin Issue / Server Blockage at our Website www.studybadshah.com then He will be send a New Link for that Particular E-Book to Access the same and if Still the Issue is Not Resolved Because of Technical Error/ Web Admin Issue / Server Blockage at our website then His Amount for that Particular Purchase will be refunded by our website via PayTM.
- All the Terms, Matters & Disputes are Subjected to "Delhi" Jurisdiction Only.

CONTENTS

Sample Preview
of
The Chapter

Published by:

NEERAJ
PUBLICATIONS
www.neerajbooks.com

ADVANCED DISCRETE MATHEMATICS

RECURRENCES

Recurrence Relations

1

INTRODUCTION

Earlier, we have studied to solve different types of combinatorial problems using various tools. However, there are still many other problems involving counting which cannot be solved only with the techniques we have learnt till now. For example, consider the problem of counting number of binary strings of length n that do not have two consecutive ones (or zeros).

Let us denote the number of such binary strings as a_n, then $a_1 = 2$ and $a_2 = 3$. Also, it can be shown that $a_n = a_{n-1} + a_{n-2}$ for $n \geq 2$. This is an example of a recurrence relation. Here it relates the values of a_n, a_{n-1} and a_{n-2}. We can, in fact, find an explicit expression for a_n using this relation. In this chapter, we will study how to formulate such recurrence relations for solving combinational problems. Introduced here are recurrence relations of three well-known examples, the Fibonacci Recurrence, Towers of Kashi (or Hanoi) and the number of ways of parenthesising an expression.

Later, we take up more examples on formulating the recurrences.

A formal definition of recurrence relation then follows alongwith terminology related to recurrences e.g. order and degree of a recurrence relation.

Finally, we analyse the Divide and Conquer techniques used in the design of algorithms that lead to recurrences in natural way. The focus is on recurrences associated with algorithms for locating the maximum and minimum elements of a list, quick multiplication of integers etc. Topics covered in this chapter are:

1. Definition of a recurrence relation;
2. Examples of recurrence relations;
3. Setting up of recurrence relations;
4. Writing recurrences for divide and conquer algorithms.

CHAPTER AT A GLANCE

THREE RECURRENT PROBLEMS

We start our study of recurrence relations with three interesting problems that provide a good idea of the subject-matter. The first two of these problems are famous and popular. All three have a solution based on the concept of recurrences. This is because the solution to each of them depends on the solution of smaller instances of the same problem.

Problem 1: Rabbits and the Fibonacci Numbers *n*: The problem of breeding rabbits was originally put forward by **Leonardo di Pisa,** better known as **Fibonacci.** He posed it in 1202 in his book **Liber abaci.** The problem is one pair of rabbits, one male and one female, live on an island. They start breeding at the end of two months and produce a pair of rabbits of opposite sex at the end of each month thereafter. Suppose f_n is the number of pairs of rabbits after n months. Then $f_1 = 1$. The rabbits start breeding only after two months and the young ones will be born one month afterwards. Hence, young ones appear only at the end of third month. Clearly, the number of pairs of rabbits is still 1 at the end of the second month i.e $f_2 = 1$. At the end of the third month, the pair would have given birth to one more pair. Table 1 gives further details. To know the number of pairs after n months, we have to add the number of pair after $n - 1$ months to the number of pairs born in the nth month. However, the newborns come from pairs at least two months old, i.e. from the pairs that are already there after $n - 2$ months; there are f_{n-2} of these. Hence, the sequence $\{f_n | n \geq 1\}$ satisfies the condition $f_n = f_{n-1} + f_{n-2}$ if $n \geq 3$. These f_n are called **Fibonacci numbers.**

Table 1: Number of Rabbits on the Island

Months	Reproducing Pairs (at least two months old)	Young Pairs (not more than two months old)
1		
2		
3		
4		
5		
6		

Obviously, we have not yet solved the problem. However, we have a uniquely defined sequence describing its future members in term of present members. We may also define f_n as a function of n. (See Q. 1)

We will take up the Fibonacci sequence again later, but in the meantime we consider another famous recurrence problem.

Problem 2: The Tower of Kashi (or Hanoi)

This problem was first stated by French mathematician Edouard Lucas in 1883. There is a tower of eight discs, initially stacked in decreasing size on one of three pegs as shown in Figure 1.

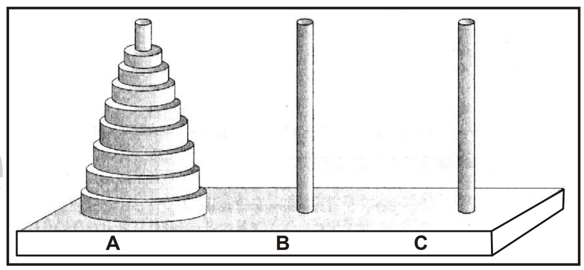
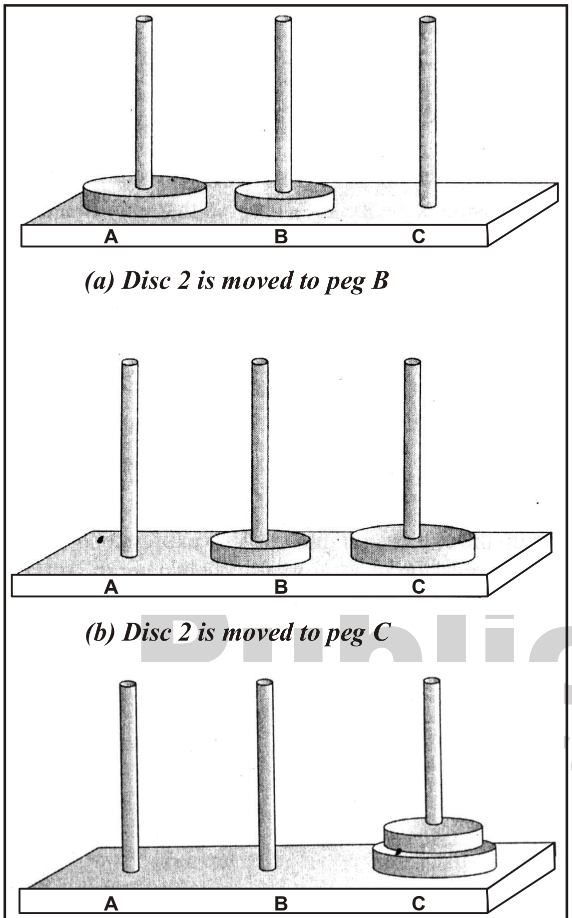


Fig. 1: Initial position of the towers of Kashi (Hanoi) problem

The objective is to transfer the complete tower to one of the other two pegs, moving only one disc at a time without at any time moving a larger disc onto a smaller one. Lucas embellished this tower with a legend about a much bigger **Tower of Brahma,** which has 64 discs of gold resting on three diamond needles. "At the beginning of time", he wrote , "God placed these golden diamond needles", "God placed these golden discs on the first needle and said that a group of priests should transfer them to a third, according to the rules above. The Tower will crumble and the world will come to an end once that task is finished."

We now generalise the problem and check the results for n discs. Let T_n be the minimum number of moves that will transfer n discs from one peg to another satisfying the given rules. Obviously, $T_1 = 1$, and $T_2 = 3$ (Figure 2). We can first experiment with two disks to find a general strategy: We first transfer the $n - 1$ smallest discs to B (using T_{n-1} moves), then move the largest (needing one move) to C. A is now empty and we can use it to transfer the discs on peg B to C. In this way, we can transfer n discs.



(a) Disc 2 is moved to peg B

(b) Disc 2 is moved to peg C

(c) Disc 2 is moved to peg C

Fig. 2: The moves involved in transferring 2 discs

(for $n \geq 2$) in at most $2T_{n-1} + 1$ moves. Hence, $T_n \leq 2 T_{n-1} + 1$, if $n \leq 2$. We have used "$\leq$" instead of " = " here. This is because our strategies proves only that $2T_{n-1} + 1$ move are sufficient but it does not prove that we can transfer the discs with lesser number of moves. In fact, it is not possible to do so. At some point, we have to move the largest disc. At that time, the $n - 1$ smallest must be on a single peg because the largest disc is on one peg and it is now being moved to the third empty peg to be at its bottom-most position. It has taken at least T_{n-1} moves to put them on that peg. After moving the largest disc for the last time, we have to now transfer the $n - 1$ smaller discs (which are again on a single peg) back onto the largest disc. This also needs T_{n-1} moves. Therefore, $T_n \geq 2T_{n-1} + 1$ if $n \geq 2$. Both the inequalities, $T_n \geq 2\,T_{n-1} + 1$ and $T_n \leq 2\,T_{n-i} + 1$ can be true only if $T_n = 2T_{n-1} + 1$.

As a matter of fact, the priests of Varanasi will require a minimum of $2^{64} - 1 = 18446\ 744\ 073\ 709\ 551\ 615$ moves to transfer the golden discs. At the superfast rate of one move per second, they will need more than 5×10^{11} years to complete the task! The doomsday for us is quite distant!

Example 1: Suppose two candidates A and B get the same number of votes, n each, in an election. The counting of votes is mostly carried out in an arbitrary order and as such during the counting A may lead for sometime and B may lead for sometime. The number of ways to count the votes such that A does not lag behind B at any time is the nth Catalan number. We can represent a vote for A by + and a vote for B by –. With this the nth Catalan number is the number of sequences of pluses and minuses such that the number of pluses are equal to number of minuses at any stage of the sequence. We can call such a sequence an admissible sequence. We now consider a special case where 8 votes are polled, 4 for A and 4 for B. One voting sequence where A never trails is (+, –, +, +, –, –, +, –). If we omit last two terms, we have the sequence (+, –, +, +, –, –) with 3 pluses and 3 minuses i.e. there are at least as many pluses as there are minuses. If the last term is not considered, we obtain (+, –, +, +, –, –, +), where there are 4 pluses and 3 minuses, i.e. pluses are more than minuses.

Example 2. In computer science a data structure called **Stack** is often used. A stack is a list which can be changed by insertions or deletions at its top. An insertion is termed a **push** and a deletion is termed a **pop**. A sequence of pushes and and pops of length $2n$ is said to be admissible if there are n pushes and n pops and at each stage of the sequenc there occur at least as many pushes as pops. Let there be a string 123...n of the set N = {1, 2, 3, ..., n} and an admissible sequence of pushes and pops of length $2n$. Each push in the sequence moves the last element in the input string to the stack and every pop moves the element on the top of the stack to the beginning of the output string. After doing n pushes and pops, the output string is permutation of N called a **stack permutation**. The number of stack permutations of 123... n is the nth Catalan number. Suppose, we represent a pop by a and a + push by a –. Clearly, every admissible sequence of pops and pushes relates to an admissible sequence of pluses and minuses. Let n = 4 and an admissible sequence of pops and pushes as (+, –, +, +, –, –, +, –). The stack permutation corresponding to this is shown in Table 2.

Table 2: Stack permutation corresponding to admissible sequence (+, –, +, +, –, –, +, +, –)

Sequence	Input String	Stack	Output String
+ (Push 4)	123	[4]	Empty
– (Pop 4)	123	[ ]	4
+ (Push 3)	12	[3]	4
+ (Push 2)	1	[23]	4
– (Pop 2)	1	[3]	24
– (Pop 3)	1	[ ]	324
+ (Push 1)	Empty	[1]	324
– (Pop 1)	Empty	[ ]	1324

Hence, the permutation derived is 1324. It is a stack permutation of size 4.

We find that in all the three examples discussed, we can express the nth term of a sequence in terms of one or more previous terms and a function of n. This provides us with a method to calculate the terms of the sequence accurately, though it may take sometime. When the relation between the terms is in a fair form, we can even obtain the recurrence. i.e. express the nth term as a function of n. We will soon learn how to solve the three recurrences and get their function.

MORE RECURRENCES

We have so far studied a variety of recurrent problems. It is now time to take a close look at setting up recurrence relations for combinational problems. When we trying to determine recurrence, we in fact, try to describe the counting inductively. In general, the recurrence relation leads to an alternate method of solution.

Example: Suppose C_n is the number of comparisons required to sort a list of n integers. We want to find a recurrence relation for C_n. We first locate the minimum of the n elements. This is going to be the first element of the list. We compare the first two elements and obtain the smaller among the two. We then compare it with the third element, and so on. To know the minimum of n elements we must make n–1 comparisons. As an example, if we want to know the minimum of the list 6, 5, 7, 4, 8 four comparisons would be needed as shown in Figure 3.

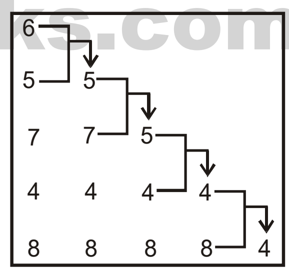


Fig. 3: Comparisons for finding the minimum of 6, 5, 7, 4, 8

We now put the minimum element found as the first element of the list. Thus we continue to sort the remaining n – 1 elements with C_{n-1} comparisons and append it after first element. Hence, $n - 1 + C_{n-1}$ comparisons have to done to sort a list of n elements.

or $C_n = C_{n-1} + n - 1$