



PROBLEM SOLVING AND PROGRAMMING

M.C.S.-11

**Chapter Wise Reference Book
Including Solved Sample Papers**

By: Nishant Gupta

Based on

I.G.N.O.U.

& Various Central, State & Other Open Universities



**NEERAJ
PUBLICATIONS**

(Publishers of Educational Books)



Retail Sales Office:

1507, 1st Floor, Nai Sarak, Delhi - 6

Mob.: 8510009872, 8510009878

E-mail: info@neerajbooks.com

Website: www.neerajbooks.com

MRP ` 280/-

Published by:

NEERAJ PUBLICATIONS

Sales Office : 1507, 1st Floor, Nai Sarak, Delhi-110 006

E-mail: info@neerajbooks.com

Website: www.neerajbooks.com

Reprint Edition with Updation of Sample Question Papers Only | Typesetting by: Competent Computers | Printed at: Novelty Printer

Notes:

1. For the best & up-to-date study & results, please prefer the recommended textbooks/study material only.
2. This book is just a Guide Book/Reference Book published by NEERAJ PUBLICATIONS based on the suggested syllabus by a particular Board/University.
3. The information and data etc. given in this Book are from the best of the data arranged by the Author, but for the complete and up-to-date information and data etc. see the Govt. of India Publications/textbooks recommended by the Board/University.
4. Publisher is not responsible for any omission or error though every care has been taken while preparing, printing, composing and proof reading of the Book. As all the Composing, Printing, Publishing and Proof Reading, etc. are done by Human only and chances of Human Error could not be denied. If any reader is not satisfied, then he is requested not to buy this book.
5. In case of any dispute whatsoever the maximum anybody can claim against NEERAJ PUBLICATIONS is just for the price of the Book.
6. If anyone finds any mistake or error in this Book, he is requested to inform the Publisher, so that the same could be rectified and he would be provided the rectified Book free of cost.
7. The number of questions in NEERAJ study materials are indicative of general scope and design of the question paper.
8. Question Paper and their answers given in this Book provide you just the approximate pattern of the actual paper and is prepared based on the memory only. However, the actual Question Paper might somewhat vary in its contents, distribution of marks and their level of difficulty.
9. Any type of ONLINE Sale/Resale of "NEERAJ IGNOU BOOKS/NEERAJ BOOKS" published by "NEERAJ PUBLICATIONS" on Websites, Web Portals, Online Shopping Sites, like Amazon, Flipkart, Ebay, Snapdeal, etc. is strictly not permitted without prior written permission from NEERAJ PUBLICATIONS. Any such online sale activity by an Individual, Company, Dealer, Bookseller, Book Trader or Distributor will be termed as ILLEGAL SALE of NEERAJ IGNOU BOOKS/NEERAJ BOOKS and will invite legal action against the offenders.
10. Subject to Delhi Jurisdiction only.

© Reserved with the Publishers only.

Spl. Note: This book or part thereof cannot be translated or reproduced in any form (except for review or criticism) without the written permission of the publishers.

Get Books by Post (Pay Cash on Delivery)

If you want to Buy NEERAJ BOOKS for IGNOU Courses then please order your complete requirement at our Website www.neerajbooks.com where you can select your Required NEERAJ IGNOU BOOKS after seeing the Details of the Course, Name of the Book, Printed Price & the Cover-pages (Title) of NEERAJ IGNOU BOOKS.

While placing your Order at our Website www.neerajbooks.com You may also avail the Various "Special Discount Schemes" being offered by our Company at our Official website www.neerajbooks.com.

We also have "Cash on Delivery" facility where there is No Need To Pay In Advance, the Books Shall be Sent to you Through "Cash on Delivery" service (All The Payment including the Price of the Book & the Postal Charges etc.) are to be Paid to the Delivery Person at the time when You take the Delivery of the Books & they shall Pass the Value of the Goods to us. We usually dispatch the books Nearly within 3-4 days after we receive your order and it takes Nearly 4-5 days in the postal service to reach your Destination (In total it take nearly 8-9 days).



NEERAJ PUBLICATIONS

(Publishers of Educational Books)

1507, 1st Floor, NAI SARAK, DELHI - 110006

Mob.: 8510009872, 8510009878

E-mail: info@neerajbooks.com Website: www.neerajbooks.com

CONTENTS

PROBLEM SOLVING AND PROGRAMMING

Question Bank – (Previous Year Solved Question Papers)

<i>Question Paper—Exam Held in February-2021 (Solved)</i>	1-9
<i>Question Paper—June, 2019 (Solved)</i>	1-8
<i>Question Paper—June, 2018 (Solved)</i>	1-7
<i>Question Paper—June, 2017 (Solved)</i>	1-11
<i>Question Paper—June, 2016 (Solved)</i>	1-9
<i>Question Paper—June, 2015 (Solved)</i>	1-9
<i>Question Paper—June, 2014 (Solved)</i>	1-8

<i>S.No.</i>	<i>Chapterwise Reference Book</i>	<i>Page</i>
1.	Problem Solving	1
2.	Basics of C	5
3.	Variables and Constants	11
4.	Expressions and Operators	16
5.	Decision and Loop Control Statements	22
6.	Arrays	40
7.	Strings	49
8.	Functions	59

<i>S.No.</i>	<i>Chapter</i>	<i>Page</i>
9.	Structures and Unions	72
10.	Pointers	77
11.	The C – Preprocessor	92
12.	Files	97
●	Some More Important Questions	108
●	Short Answer Type Questions Miscellaneous	125
●	Important ‘C’ Programs	130
●	Appendix-I : ASCII Values of Characters	150
●	Appendix-II : ANSI ‘C’ Library Functions	151
●	Appendix-III : Glossary	155
●	<i>Sample Question Paper</i>	159
		■ ■

**Sample Preview
of the
Solved
Sample Question
Papers**

Published by:



**NEERAJ
PUBLICATIONS**

www.neerajbooks.com

QUESTION PAPER

Exam Held in
February – 2021

(Solved)

PROBLEM SOLVING AND PROGRAMMING

M.C.S.-11

Time: 3 Hours]

[Maximum Marks: 100

Note : Question no. 1 is compulsory. Attempt any three questions from the rest.

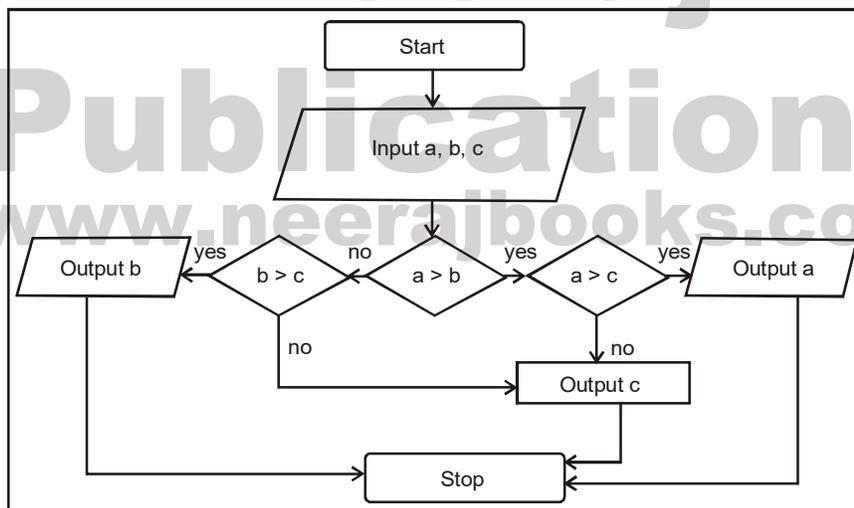
Q. 1. (a) Write an algorithm and draw corresponding flowchart to find the largest number among 3 numbers given as input.

Ans. Algorithm :

- Step 1 : Start
- Start 2 : Input a, b, c
- Start 3 : if $a > b$ goto step 4, otherwise goto step 5

- Start 4 : if $a > c$ goto step 6, otherwise goto step 8
- Start 5 : if $b > c$ goto step 7, otherwise goto step 8
- Start 6 : Output "a is the largest", goto step 9
- Start 7 : Output "b is the largest", goto step 9
- Start 8 : Output "c is the largest", goto step 9
- Start 9 : Stop

Flowchart :



(b) Explain the conditional operator in C with the help of an example. Compare it with if...else statement.

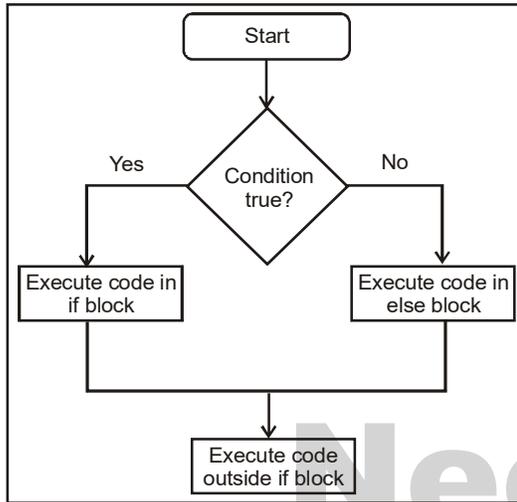
Ans. Conditional Operator: C provides an called as the conditional operator which is closely related to the if/else structure. The conditional operator is C's only ternary operator – it takes three operands. The operands together with the conditional operator form a conditional expression. The first operand is a

condition, the second operand represents the value of the entire conditional expression if the condition is true and the third operand is the value for the entire conditional expression if the condition is false. The syntax is as follows: (condition)? (expression1): (expression2); If condition is true, expression1 is evaluated else expression2 is evaluated.

Let us see the following examples: (i) $x = (y = 50 ? \text{"Passed"} : \text{"failed"})$; The above statement will print "passed" grade ≥ 50 else it will print "failed" (ii)

(a>b) ? printf (“a is greater than b\n”); printf (“b is greater than a\n”); If a is greater than b, then first printf statement is executed else second printf statement is executed.

The If-Else statement



The if-else is statement is an extended version of If. The general form of if-else is as follows:

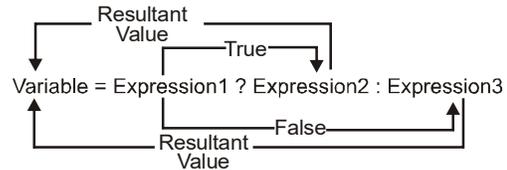
```

if (test-expression)
{
  True block of statements
}
Else
{
  False block of statements
}
Statements;
  
```

In this type of a construct, if the value of test-expression is true, then the true block of statements will be executed. If the value of test-expression if false, then the false block of statements will be executed. In any case, after the execution, the control will be automatically transferred to the statements appearing outside the block of If.

The conditional operator is kind of similar to the if-else statement as it does follow the same algorithm as of if-else statement but the conditional operator takes less space and helps to write the if-else statements in the shortest way possible.

Conditional or Ternary Operator (?) in C/C++



Syntax:

The conditional operator is of the form variable = Expression1 → Expression2 : Expression3

It can be visualized into if-else statement as:

```

if(Expression1)
{
  variable = Expression2;
}
else
{
  variable = Expression3;
}
  
```

Since the Conditional Operator ‘?:’ takes three operands to work, hence they are also called **ternary operators**.

Working:

Here, **Expression1** is the condition to be evaluated. If the condition(**Expression1**) is True then **Expression2** will be executed and the result will be returned. Otherwise, if the condition (**Expression1**) is false then **Expression3** will be executed and the result will be returned.

(c) Explain the following user-defined data types:

(i) typedef

Ans. typedef is used to define new data type names to make a program more readable to the programmer. The main use for typedef seems to be defining structures. For example: typedef struct {int age; char *name} person; person people; Take care to note that person is now a type specifier and NOT a variable name.

(ii) enum

Ans. It is used to assign names to the integral constants which makes a program easy to read and maintain. The keyword “enum” is used to declare an enumeration. Here, is the syntax of enum in C language, enum enum_name{const1, const2, }

Sample Preview of The Chapter

Published by:



**NEERAJ
PUBLICATIONS**

www.neerajbooks.com

PROBLEM SOLVING AND PROGRAMMING THROUGH C

Problem Solving

1

1.1 INTRODUCTION AND OBJECTIVES

If we think about our daily life, apart from computer, we encounter many problems and also solve them, some of them have a proper solution already defined, for same we have to think ourself and find one. The basic path to problem solving remains that first identify the goal, secondly make use of available resources and finally use them to reach the goal. But in this chapter we use INS concept of problem solving pertaining to computer programming.

When we talk of problem solving pertaining to computer programming, we take into consideration, the output required as goal and input data available as resource and other parametres as time constraints etc.

There is no standard or universal approach to solve a problem. Same problem can be solved in different ways, we cannot say which one is correct. Both ways would we correct if they produce correct output. The problem solving skills are integral part of computer programming. No matter what language we are using.

The approach to problem solving comes with experience and practice. Problem solving skills are equally important in project life-cycles like study, designing, development, testing and implementation.

The computer problem solving requires:

- (i) Analysis of problem.
- (ii) Identify inputs/outputs.
- (iii) Plan a process with inputs available.

Problem solving requires personal creativity, and analytical ability, which comes by practice and exposure.

Computer is used to solve the problems. This machine follows the set as stored instructions called program very quickly and have the memory to store data. The computer cannot think, it only executes program instructions very quickly.

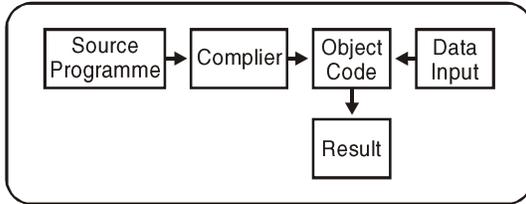
Computers do not work like humans. When computer is used to solve a problem, we must specify the input required, the process to be done and the output required. Once these are considered an algorithm and a program is developed. It is not necessary the programme developed gives the correct result. If it does not give the correct result we need to find the errors and then correct them, this process is called Debugging.

All types of computer programs are referred to as software. Physical parts of equipment such as electronic parts, devices, come under hardware. Software works on the hardware and uses it to have the output. Sometimes operations performed by the software are built in hardware.

2 / NEERAJ : PROBLEM SOLVING AND PROGRAMMING THROUGH C

Set of instructions of the high level language used to code a problem and to find a solution is referred to as source programme. A software is used to read that code and translate into machine language which computer understands is used, which is known as compiler or interpreter.

The following is the path to from making a program and execute it.



1.2 PROBLEM SOLVING TECHNIQUES

Problem solving comes by experience, it is a creative process which defines the systematic process. The number of steps can be taken to solve the problem is the performance of the algorithm.

1.2.1 Steps for problem solving

(i) **Problem Definition:** The problem should be defined and the clear concept should be developed that what must be done, rather than how is it done.

(ii) **Start analysing problem:** There are many ways to solve a problem. What comes in your mind may be correct but the approach should be of best way.

(iii) **Use specific examples:** We can take the already solved problems to the new one which save times and efforts.

(iv) **Similarities among problems :** It is important to see if there are any similarities between the current problem and the past problem which we have solved. The more experience a person have in solving problems.

(v) **Working backwards from the Solution:** Sometimes we already have the solution to the problem and then try to work backwards to the starting point. Even a guess at the solution to the may be enough.

1.2.2 Using computer as a problem solving tool.

The computer is used to solve problem only when the problem's solution is made in step by step manner using algorithm.

follow the steps

- (i) Develop Algorithm and a flow chart of the sloved problem.
- (ii) write a computer program using programming language.
- (iii) Enter the program using editor.
- (iv) Compile the program
- (v) Test and Debug the program
- (vi) Run and Get the Results.

1.3 DESIGN OF ALGORITHMS

The first step in program development is to describe what you want the computer to do. This plan expressed as a sequence of jobs is called algorithm. An algorithm of a programming language becomes program.

Definition: An algorithm is a set of well-defined rules for the solution of a problem in a finite number of steps. An algorithm can be expressed in pseudo codes—something resembling the programming language.

Features of Algorithm:

(i) **Proper understanding of the problem:** The problem should be properly understood in order to develop an effective algorithm.

(ii) **Use of procedures and Subroutines:** When the problem becomes complex the algorithm written should be in modular manner *i.e.* one main problem to be solved using many small problem.

(iii) **Choice of variables:** Proper variables names should be used in algorithm so that the name of variable gives the meaning to use it.

(iv) **Documentation of program:** If the algorithm is complex then it should have comments at appropriate places so that the other person can understand the code.

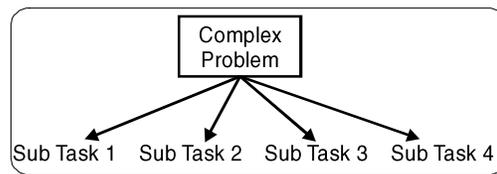
Some examples of algorithm.

1. Algorithm to calculate factorial of a number.

1. Start
2. Input numbr n
3. Initialize $i = 1, f = 1$
4. Repeat steps 4 – 6 until $i = n$
5. $f = f * i$
6. $i = i + 1$
7. print f
8. Stop

1.3.4 Top Down Design

A technique for algorithm design that tries to accomodate human limitation is known as top down design. Top down design provieds the way of handling the logical complexity and detail encountered in computer algorithms. In this approach the complex problem is divided into many simple sub problems and each of them is worked upon and then the solution is combined.



1.4 ANALYSIS OF ALGORITHM EFFICENCY

Algorithms use some of the computers resources like CPU time, memory, I/O etc. Because of high cost of computing, it is desirable to design algorithms that are economical in the use of CPU, time & memory.

- (i) Analysis can be more reliable.
- (ii) It helps in choosing a solution from many solutions available to a problem.

1.5 ANALYSIS OF ALGORITHM COMPLEXITY

The following are the qualities of the algorithm:

- (i) Easily modifiable.
- (ii) They should be correct and clearly defined.
- (iii) Require less time, storage, I/O time.
- (iv) They are well documented.
- (v) They are not dependent on language.
- (vi) They are able to use the sub procedures.

(1) Computational complexity: More computing resources are needed to solve larger problems in the same class. Very small problems can be solved with an algorithm that exhibit exponential behaviour.

(2) The order Notation: The o-notation gives an upper bound to a function within a constant factor. For a given function $g(n)$ we denote by $o(g(n))$.

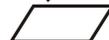
For example a double nested loop structure yields $o(n^2)$ upper bound on the worst case time.

(3) Worst and average case behaviour: In many applications it is important to have a measure of the expected complexity. In linear search the worst case will be to examine n values and average case will be to examine $n/2$ values.

1.6 FLOWCHARTS

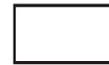
Flowcharts are used in programming to diagram the path in which information is processed through a computer to obtain the desired results. Flowchart is a graphical representation of an algorithm. It makes use of symbols which are connecting among them to indicate the flow of information & processing. It will show the general outline of how to solve a problem or performing a task. It is prepared for better understanding of the algorithm.

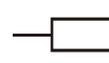
Basic symbol used in flowchart design:

-  Start/Stop
-  Question Decision (used in branching)
-  Input/Output

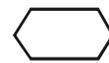
 Lines or arrows represent the direction of the flow of control

 Connector (connect one part of the flowchart to another)

 Process, Instruction

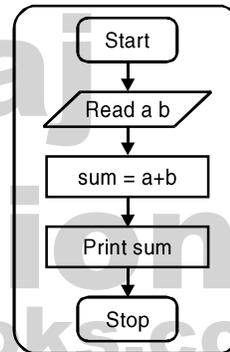
 Comments, Explanations, Definitions

Additional symbols related to more advanced programming:

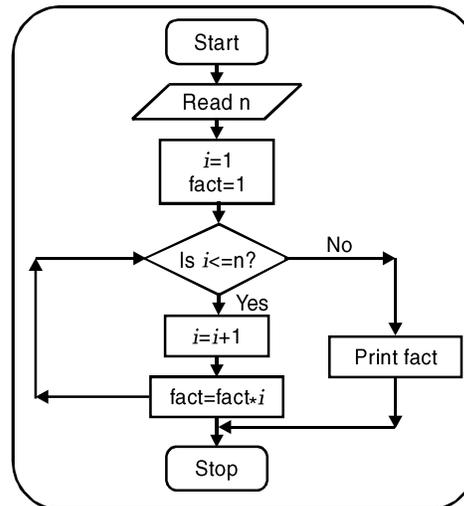
 Preparation (may be used with "do loops")

 Refers to separate flowchart

Example 1: A flowchart to display the sum of two numbers



Example 2: Flowchart to find factorial of a given numbers.



CHECK YOUR PROGRESS

Q. 1. Differentiate between flowchart and algorithms.

Solution: Algorithms.

1. An algorithm consists of a set of explicit and unambiguous finite steps which when carried out for a given set of initial conditions produce the desired output and terminate in a fixed amount of time. In other words, Algorithm is a process of breaking the given problem into smaller, simpler and interlinked steps arranged sequentially in order to achieve the desired objective.

2. The algorithm is written in English or English like language known as pseudocode. This language is not dependent on any programming language.

Flow chart

1. Flow chart is the pictorial representation of an algorithm.

2. Pictures always speak too much than any paragraph. Flowchart gives the clear idea of the process to solve the problem.

3. Some specified geometric shape boxes are used for the operations and they are connected by arrows to control the flow. Any error in the logic of the process can be detected easily.

4. A flowchart basically is the plan to be followed when the program is written. It acts like a road map for a programmer and guides him in proceeding from the starting point to the final point, while writing a computer program.

Q. 2. Compute and print the sum of a set of data values.

Solution:

Algorithm to calculate sum of set of values.

1. Start
2. Set the sum variable to zero.
3. Input the data values as long as they exist and add them to sum.
4. Display the value of sum.
5. Stop

Q. 3. Write the steps that are suggested to facilitate the problem solving process using computer.

Solution:

1. Define the problem
2. Analyse the problem for input and output required.
3. Formulate a mathematical and logical model to solve the problem.
4. Develop an algorithm to solve the problem using the model in step 3. Each step should be so simple that it can be converted into equivalent statement in programming language.

5. Design a flowchart.

6. Write a program code of the same.

7. Test the program for errors.

Q. 4. Draw an algorithm and flowchart to calculate the roots of quadratic equation $Ax^2 + Bx + c = 0$.

Solution:

Algorithm

1. Start
2. Input the value of A, B and C.
3. Calculate $D = B * B - 4 * A * C$
4. Check if $(D < 0)$ print (Roots are imaginary not real)
5. if $(D > 0)$ Print+(Roots are Real)
 - (a) $x_1 = -B / (2 * A)$
 - print (x_1)
6. If $(D > 0)$ print (Roots are Real and distinct)
 - (a) $x_1 = (-B + D) / (2 * A)$
 - (b) $x_2 = (-B - D) / (2 * A)$
 - (c) Print (X_1, X_2)
7. Stop.

Flowchart:

