Source: www.geeksforgeeks.org/best-way-to-start-with-competitive-programming-geeksforgeeks-cp-live-course/

22TP106 PROBLEM SOLVING THROUGH PROGRAMMING - II

Hours Per Week :

L	T	P	C
1	2	2	3

PREREQUISITE KNOWLEDGE: Programming in C.

COURSE DESCRIPTION AND OBJECTIVES:

This course is aimed to impart knowledge on some advanced concepts of C programming language and problem solving. It covers pointers, structures, unions and file manipulations. At the end of this course, students will be able to design, implement, test and debug complex problems using features of C.

MODULE-1

UNIT-1 **4L+8T+8P=20 Hours**

Pointers: Declaration, Initialization, Multiple indirection, Pointer arithmetic, Relationship between arrays and pointers

Problem Solving on Global Coding Platform using Data types and operators; Control statements; functions

UNIT-2 **4L+8T+8P=20 Hours**

POINTERS: Scaling up - array of arrays, array of pointers, pointer to a pointer and pointer to an array; Dynamic memory allocation functions.

Problem Solving on Global Coding Platform using Data types and operators; Control statements; functions.

PRACTICES:

1. a. What is a pointer in C, and why is it important in C programming?

   b. A program to declare a pointer, assign a value, and print its value.
2. a. Explain the difference between the address-of operator (&) and the dereference operator (*) in C.

   b. A program to demonstrate pointer arithmetic, e.g., adding to and subtracting from pointers.
3. a. Explain the concept of a double pointer (pointer to a pointer).

   b. A program showing the difference between pointers to constants and constant pointers.
4. a. How can you use pointers to dynamically allocate memory in C?

   b. A program using malloc to allocate memory for an array and then freeing it with free.
5. a. What is the purpose of a null pointer?

   b. A program using null pointers and checking for null before dereferencing.

MODULE-2

UNIT-1 **4L+8T+8P=20 Hours**

STRUCTURES, UNIONS & FILES

Structures: Defining a structure, Declaring structure variable, Operations on structures, Pointers to structure - declaring pointer to a structure, accessing structure members using pointer; Array of structures, Nested structures, Passing structures to functions - passing each member of a structure as a separate argument, passing structure variable by value, passing structure variable by reference/address; Typedef and structures.

UNIT-2 **4L+8T+8P=20 Hours**

UNIONS: Defining a union - declaring union variable, operations on union; Pointers to union - declaring pointer to a union, accessing union members using pointer; Array of union, Nested union, Typedef and union, Enumerations, Bit-fields.

Files: Introduction to files, Streams, I/O using streams – opening a stream, closing stream; Character input, Character output, File position indicator, End of file and errors, Line input and line output, Formatted I/O, Block input and output, File type, Files and command line arguments.

PRACTICES:

Write a C program to create a struct, named Student, representing the student's details as follows: first_name, last_name, Age and standard.

Example

Read student data

john

carmack

15

10

Display the data in the following format

First Name: john

Last Name: carmack

Age: 15

Standard: 10

2. Declare a structure POINT. Input the coordinates of point variable and write a C program to determine the quadrant in which it lies. The following table can be used to determine the quadrant.

Quadrant	X	Y
1	Positive	Positive
2	Negative	Positive
3	Negative	Negative
4	Positive	Negative

Example

Input the values for X and Y coordinate: 7 9

The coordinate point (7,9) lies in the First quadrant.

3. Bob and Alice both are friends. Bob asked Alice how to store the information of the books using Structures. Then Alice written a c program to store the information of books using book structure by taking different attributes like book_name, author, book_id, price. Write a C program to read and display the attributes of the books using structures.

Sample Input:

Enter number of books: 1

Enter the book name: c Programming

Enter the author name: balaguruswamy

Enter the book ID: 23413

Enter the book price: 500

Sample Output:

The details of the book are:

The book name is: c Programming

SKILLS:

- ✓ *Analysis of the problem to be solved.*
- ✓ *Application of various file operations effectively in solving real world problems.*
- ✓ *Develop C programs that are understandable, debuggable, maintainable and more likely to work correctly in the first attempt.*

The author name is: balaguruswamy

The book ID is: 23413

The book price is: 500.00

4. Ramesh wants to do addition on complex numbers. He did it with regular practice but Charan asked him to do with the help of structures by following below Criteria.

   Write a C program that defines a structure named 'Complex' consisting of two floating point members called "real and imaginary". Let c1 and c2 are two Complex variables; compute the sum of two variables.

   Example:

   c1= 2 8

   c2= 6 4

   Sum= 8.000000+12.000000i

5. Customer Payment Details is a structure with members as customers_name, address, account_number, payment_status(paid(1)/ not_paid(0)), due_date, and amount. In this example, payment_date is another structure with month, day and year as integer members. So, every customer record can be considered as an array of structures.

   Write a C program that displays the amount to be paid by each customer along with their names. If payment_status is 1, display NIL for such customers.

   Input Format:

   First line of input contains 'n' number of customers, followed by 8 lines of input for each customer. Each line represents (customers_name, address, account_number, amount payment_status(paid(1)/ not_paid(0)), and due_date).

   Output Format: First line of output is Amount to be paid by each customer as on date: followed by n lines of output. Each line contains name of the customer followed by tab space, and amount to be paid.

   Hint: Use nested structure to represent date.

6. Write a 'C' program to accept customer details such as: Account_no, Name, Balance using structure. Assume 3 customers in the bank. Write a function to print the account no. and name of each customer whose balance < 100 Rs.

7. Write a C program to accept details of 'n' employee(eno, ename, salary) and display the details of employee having highest salary. Use array of structure.

8. Write a C program to print the bill details of 'N' number of customers with the following data: meter number, customer name, no of units consumed, bill date, last date to deposit and city. The bill is to be calculated according to the following conditions:

No. of units	Charges
For first 100 units	Rs.0.75 per unit
For the next 200 units	Rs.1.80 per unit
For the next 200 units	Rs.2.75 per unit

   Sample Input

   Enter no. of customers

   1

   Enter Meter Number AP01213

   Enter Customer Name: Karthik

   Enter No. of units consumed: 200

   Enter Bill date:22/01/2021

   Enter Last date: 12/2/2021

   Enter City: Guntur

   Sample Output

Meter Number AP01213

Customer Name: Karthik

No. of units consumed: 200

Bill date:22/01/2021

Last date: 12/2/2021

City: Guntur

Total Amount: 255.000000

9. Write a C program that creates a student file containing {Roll No, Student Name, Address, Stream}, where the data will be inserted and display the list of students who are in CSE (Stream=CSE).

   Input: A file name

   Output: The attributes such as Roll_No, Student_Name, Stream, Address.

   Sample Input

201fa4200	Raja	CSE	Guntur
201fa4201	Bala	IT	Tenali

   Sample Output

201fa4200	Raja	CSE	Guntur

10. Write a C program that reads content from an existing text file and write the same in a new file by changing all lowercase alphabetic character to upper case. (Existing file may contain digit and special characters).

    Example:

    Input: Enter the file name.

    Output: New file with updated content.

11. Write a C program to count the occurrences of the given string in a file.

    Example:

    Input: Enter the File name to read the string to be counted.

    Output: Display the count of occurrences of the string.

12. Write a C Program to transfer the data from one location to another location without changing the order of the content.

    Example:

    Read the file name from the user. If the source file exists, Transfer the data and display the message as “Data is transferred successfully” otherwise display the message “No such file is existing in the directory.”

13. Write a C program that reads numbers and write them into a text-file. Also find odd and even numbers in that file and store it in 2 separate files named odd.txt and even.txt. All the values should be in ascending order.

    Input: Enter the values.

    Output: Creates a separate file for Even and Odd numbers.

    Sample Input:

    4 43 2 53 45

    Sample Output:

    Even.txt: 2 4

    Odd.txt: 43 45 53

14. Write a C program to replace the content in the given text file.

    Input: Enter the file name, line number to be replaced and the new content

    Output: New file with replaced lines.

    Example:

    Sample Input:

```
Enter the file name: abc.txt
Enter the line no to replace: 3
Enter the content: Files stores data presently.
```

Sample Output:

```
Line no 3 is replaced with the given content.
The content of the file abc.txt contains:
test line 1
test line 2
Files stores data presently
test line 4
```

COURSE OUTCOMES:

Upon successful completion of this course, students will have the ability to:

CO No.	Course Outcomes	Blooms Level	Module No.	Mapping with POs
1	Identify suitable data type for operands and design of expressions having right precedence.	Apply	1,2	1
2	Apply decision making and iterative features of C Programming language effectively.	Apply	1,2	1
3	Select problem specific data structures and suitable accessing methods.	Analyze	1,2	1,2
4	Design and develop non- recursive and recursive functions and their usage to build large modular programs and also able to design string manipulation functions.	Create	1,2	3
5	Develop C programs that are understandable, debuggable, maintainable and more likely to work correctly in the first attempt.	Evaluate	1,2	3,4

TEXT BOOKS:

1. Behrouz A. Forouzan, Richard F.Gilberg, “Programming for Problem Solving”, 1st edition, Cengage publications, 2019.
2. Ajay Mittal, “Programming in C - A Practical Approach”, 1st edition, Pearson Education, India, 2010.

REFERENCE BOOKS:

1. Reema Thareja, “Computer Fundamentals and Programming in C”, 1st edition, Oxford University Press, India, 2013.
2. Herbert Schildt, “C: The Complete Reference”, 4th edition, Tata McGraw-Hill, 2017.
3. R G Dromey and Pearson, “How to solve it by Computer”, 2nd Edition, 1998.Ibeaquis alique volorepro omnias dolum et pos es ma que pro doluptae porum inietur, quas minctatibus del molest odit verias imoluptaqui arum cum ius enet alique sitios est poreictint a sunt rent et experum nonsectempos at etus, quat ommos seque in comnihitatis es sequunt quam quo offictis ex eum fuga. Namus erchici endictati int as qui doloriberro odignation re et erferae. Alignatat ut atias culpa voluptatur mil et etum autassi ntotatur?