

1-Day C Programming Challenge

Goal : By the end of 21 days, you will be able to:

- Write C programs confidently
- Understand loops, arrays, functions, pointers
- Solve beginner and intermediate programming problems
- Build mini console applications

Week 1 - C Programming Basics

Day 1 - Introduction to C

Topics

- Structure of C Program
- Character Set
- Keywords

Learn

A simple C program consists of

```
#include<stdio.h>

int main()
{
    printf("Hello World");
    return 0;
}
```

Understand

- Header files
- main()
- printf()
- return statement

Example Programs

1. Hello World
2. Print your name
3. Print address
4. Print college name

Challenge

Print

```
*****  
*****  
*****
```

Day 2 - Data Types, Variables & Constants

Topics

- Data Types
- Variables
- Constants

Learn

```
int age=20;  
float salary=25000.50;  
char grade='A';
```

Example

```
#include<stdio.h>  
  
int main()  
{  
    int age=21;  
    printf("%d",age);  
}
```

Programs

- Store age
- Store marks
- Swap two numbers
- Area of rectangle

Challenge

Find area and perimeter of rectangle.

Day 3 - Operators

Topics

- Arithmetic
- Assignment
- Increment
- Decrement

Example

```
int a=10,b=3;

printf("%d",a+b);
printf("%d",a-b);
printf("%d",a*b);
printf("%d",a/b);
printf("%d",a%b);
```

Programs

- Simple calculator
- Celsius to Fahrenheit
- Fahrenheit to Celsius

Challenge : Convert days into years, months and days.

Day 4 - Input, Output & Type Conversion

Topics

- scanf()
- printf()
- Type Conversion

Example

1-Day C Programming Challenge

Goal : By the end of 21 days, you will be able to:

- Write C programs confidently
- Understand loops, arrays, functions, pointers
- Solve beginner and intermediate programming problems
- Build mini console applications

Week 1 - C Programming Basics

Day 1 - Introduction to C

Topics

- Structure of C Program
- Character Set
- Keywords

Learn

A simple C program consists of

```
#include<stdio.h>

int main()
{
    printf("Hello World");
    return 0;
}
```

Understand

- Header files
- main()
- printf()
- return statement

Example Programs

1. Hello World
2. Print your name
3. Print address
4. Print college name

Challenge

Print

```
*****
*****
*****
```

Day 2 - Data Types, Variables & Constants

Topics

- Data Types
- Variables
- Constants

Learn

```
int age=20;
float salary=25000.50;
char grade='A';
```

Example

```
#include<stdio.h>

int main()
{
    int age=21;
    printf("%d",age);
}
```

Programs

- Store age
- Store marks
- Swap two numbers
- Area of rectangle

Challenge

Find area and perimeter of rectangle.

Day 3 - Operators

Topics

- Arithmetic
- Assignment
- Increment
- Decrement

Example

```
int a=10,b=3;

printf("%d",a+b);
printf("%d",a-b);
printf("%d",a*b);
printf("%d",a/b);
printf("%d",a%b);
```

Programs

- Simple calculator
- Celsius to Fahrenheit
- Fahrenheit to Celsius

Challenge

Convert days into years, months and days.

Day 4 - Input, Output & Type Conversion

Topics

- scanf()
- printf()
- Type Conversion

Example

```
int age;

scanf("%d",&age);

printf("%d",age);
```

Programs

- Read integer
- Read float
- Add two numbers
- Average of three numbers

Challenge

Find simple interest.

Day 5 - Operators

Topics

- Relational Operators
- Logical Operators
- Bitwise Operators
- Operator Precedence

Programs

Compare two numbers

```
if (a>b)
```

Bitwise

```
5 & 3  
5 | 3  
5 ^ 3
```

Challenge

Check even/odd using bitwise operator.

Week 2 - Conditional Statements & Loops

Day 6 - Conditional Expressions

Topics

- if
- if-else
- nested if
- ternary operator

Programs

- Positive or Negative
- Largest of two numbers
- Largest of three numbers

Challenge

Check leap year.

Day 7 - Switch Statement

Programs

- Menu calculator
- Week day
- Month name
- Simple ATM menu

Challenge “ Create menu driven calculator.

Day 8 - Conditional Programs

Programs

- Grade calculation
- Voting eligibility
- Even/Odd
- Divisible by 5 and 11
- Character is vowel or consonant

Challenge

Electricity bill calculator.

Day 9 – Loops : Topics

- while
- do while
- for

Difference

while
Checks first

do while
Executes once then checks

Programs

- Print 1 to 100
- Reverse counting
- Sum of first N numbers

Challenge

Multiplication table.

Day 10 - Loop Programs

Programs

- Factorial
- Fibonacci
- Prime Number
- Palindrome
- Armstrong Number

Challenge : Perfect Number.

Day 11 - Nested Loops

Programs

Print

```
*
**
***
****
```

Print

```
1
12
123
1234
```

Challenge

Pyramid patterns.

Day 12 - Arrays

Topics : One-dimensional array

Programs

- Sum
- Average
- Largest
- Smallest

Challenge : Second largest number.

Day 13 - Arrays & Matrices

Topics

- Two-dimensional arrays
- Matrix

Programs

- Matrix addition
- Matrix subtraction
- Matrix multiplication
- Transpose

Challenge

Identity matrix.

Week 3 - Strings, Functions & Pointers

Day 14 - Strings

Topics

- String basics

Functions

`gets()`
`puts()`
`fgets()`

Programs

- Print string
- Length
- Reverse

Challenge

Palindrome string.

Day 15 - String Functions

Functions

`strlen()`
`strcpy()`
`strcat()`
`strcmp()`
`strrev()` (compiler dependent)

Programs

- Copy string
- Compare strings
- Concatenate

Challenge

Count vowels.

Day 16 - Functions

Topics

- Function
- Declaration
- Definition
- Calling

Programs

`void display()`
`int add(int,int)`

Challenge

Calculator using functions.

Day 17 - Function Programs

Programs

- Prime using function
- Factorial using function
- Fibonacci using function
- Maximum using function

Challenge

Recursive factorial.

Day 18 - Pointers

Topics

- Pointer
- Address Operator
- Dereference

Example

```
int a=10;  
  
int *p=&a;  
  
printf("%d", *p);
```

Programs

- Pointer basics
- Pointer arithmetic
- Swap using pointers

Challenge : Find largest element using pointers.

Week 4 - Structures, Unions & Mini Projects

Day 19 - Structures

Topics

- Structure
- Declaration
- Initialization
- Access members

Program

```
struct Student
{
int roll;
char name[30];
float marks;
};
```

Programs

- Student record
- Employee record

Challenge

Store details of five students.

Day 20 - Union & Preprocessor

Topics

- Union
- Structure vs Union
- #define
- #include
- #ifdef
- #ifndef
- #if
- #endif

Programs

```
#define PI 3.14
```

Challenge : Area calculator using macros.

Day 21 - Final Challenge Day

Mini Projects

Choose one: **Beginner** . Student Result System

Intermediate : Bank Management System

Intermediate : Library Management System

Intermediate : Hotel Billing System

Intermediate : Employee Payroll System

Intermediate : Inventory Management System

Daily Coding Habit

Every day follow this routine:

1. Read theory (20–30 min)
2. Understand one example (20 min)
3. Type every example yourself (30 min)
4. Solve 5–10 practice problems (60 min)
5. Complete one challenge program (30–60 min)
6. Revise previous day's concepts (15 min)

Final Outcome : After completing this 21-day challenge, you should be comfortable with:

- ☒ C program structure
- ☒ Variables and data types
- ☒ Operators and expressions
- ☒ Input/output
- ☒ Decision making (`if`, `switch`)
- ☒ Loops and pattern programming
- ☒ Arrays and matrices
- ☒ Strings and string handling
- ☒ Functions (including recursion)
- ☒ Pointers and pointer arithmetic
- ☒ Structures and unions
- ☒ Preprocessor directives
- ☒ Writing complete console-based C applications