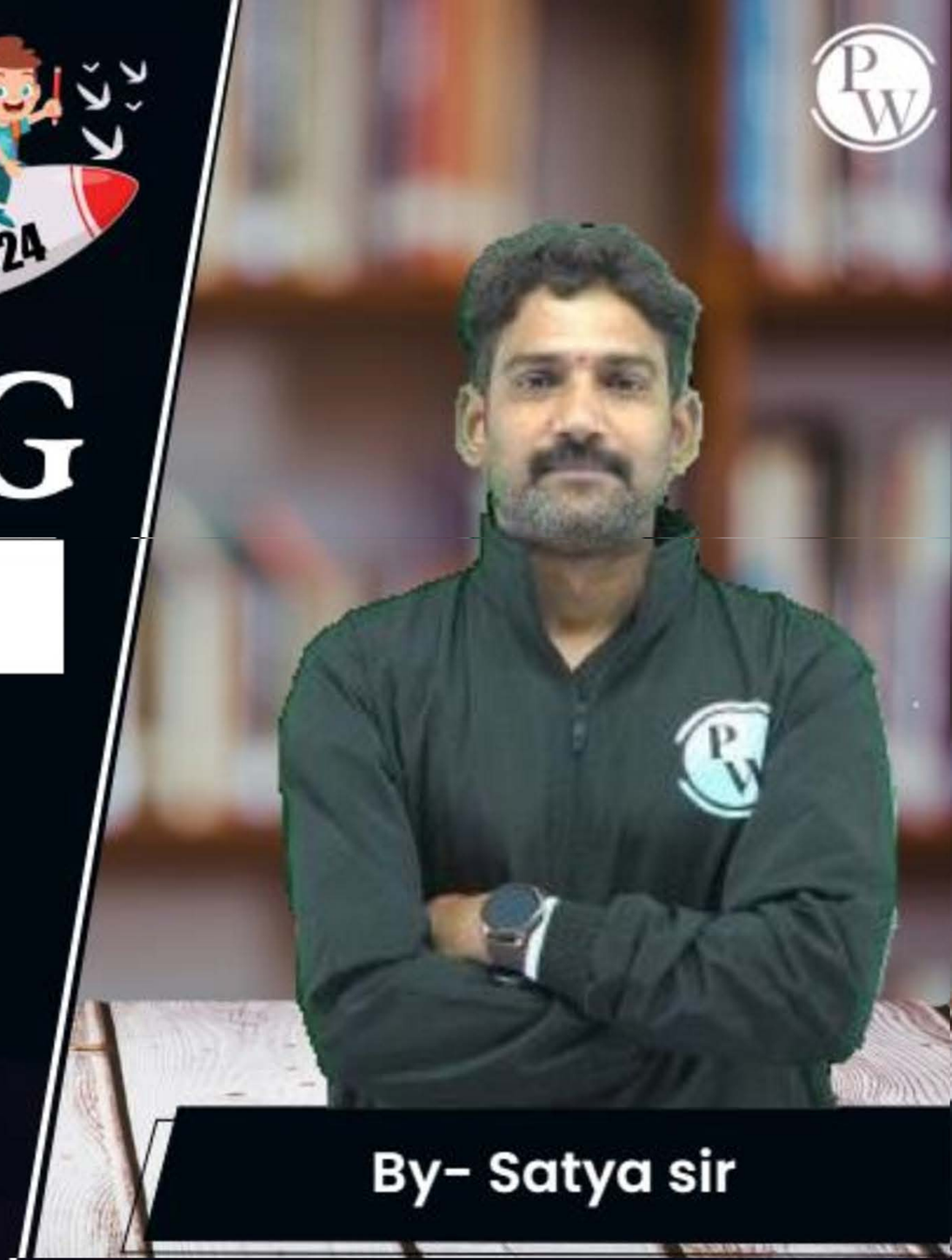
CS & IT ENGINEERING

C PROGRAMMING

Data Types and I/O Functions

Lecture No.- 01



By- Satya sir

Recap of Previous Lecture

- Relational Operators
- Assignment Operator
- Operator Precedence & Associativity

Topics to be Covered

- Data Types in 'C' Language

Topic : Data Types

Data Types

Examples of data : (Different Nature)

1) Roll Number ⇒ Whole Number ⇒ Unsigned Integer type (+ve only) (without fraction)
2) GATE_Marks ⇒ 83.666 / −3.333 / 60 / −15 ⇒ Signed float type (±ve)
3) Name ⇒ "Satya" / "Nishant" / "Tejas" / ... ⇒ String Type
4) E-mailID ⇒ "abc.123@gmail.com" ⇒ String Type
5) length, breadth ⇒ −2 / −3 / 2 / +3 ⇒ Signed Integer
6) gender ⇒ 'M' / 'F' / 'T' ⇒ Character Type

Topic : Data Types

Examples		Nature	Possible Values
date_of_month	= +ve Integer	⇒ Unsigned Integer	1 to 31
month_of_year	= +ve Integer	⇒ Unsigned Integer	1 to 12
marks_Subject	= +ve float type	⇒ Unsigned float	0 to [0.0 to 100]
Add 2 whole numbers	= ±ve Integer	⇒ Signed Integers	−∞ to +∞

Range of Values

- Small range — Less memory required
- Big Range — more memory required

Real Time Example

- No_of_Std_College = (Thousands) - Less memory
- No_of_Std_University = (Lakhs) - more Memory
- Population_Country = (Crores) - Huge Memory

Topic : Data Types

Super Market Program

Data ⇒

- Item Name
- Item Quantity
- Item brand
- Item Weight
- Item Price
- Item_in_Stock
- Discount_on_item
- Item_Category

- Employee_name
- Employee_id
- Emp_Salary
- Emp_designation
- Emp_PAN
- Emp_Aadhar
- Emp_phone No

- Customer_name
- Customer_Mobile
- Customer_Email_id

- Stock_available
- dealer_name
- dealer_Code
- TAN Number
- branch_Name
- branch_Code

Topic : Data Types

To define 1) Nature
2) Memory size (requirement)
3) Range of values for data (variables), datatype is specified.

① Data Types in 'C' : classified into 2 Types

1) Basic (or) Fundamental (or) Primary : char, int, float, void

2) Derived (or) Secondary [Extensions of Basic datatypes]
[Arrays, Pointers, Structures, Unions, Files, Enumerated datatype, typedef]

Topic : Data Types

Character DataType (Keyword : char)

- Any single Literal [Number / Symbol / alphabet]
- Memory for characters is 1 Byte (or) 8 bits
- In 'C', Characters are Processed with their unique ASCII Values.

[American Standard Code for Information Interchange]

- There are 256 ASCII values [0 to 255]

Ex:

Character	ASCII Value
'A'	65
'a'	97
'0' (zero)	48
Space	32

- Type qualifiers / Extensions :
  - Signed char (±ve)
    [char == signed char]
  - Unsigned char (+ve only)

Topic : Data Types

Integer Data Type : A whole Number, without fractional Part.

- Memory size of Integers in 'C' Language is Compiler-dependant / Processor-dependant.

  Ex: 16-bit Processor ⇒ 16-bit Compiler ⇒ Int size is 16 bits

  32-bit Processor ⇒ 32-bit Compiler ⇒ 32 bits

  64-bit Processor ⇒ 32-bit Compiler / 64-bit Compiler ⇒ 32 bits / 64-bits

- Type qualifiers / Extensions : (Let n Bytes Per Integer)

  1) Short int == Short — n/2 Bytes
  2) long int == long — 2*n Bytes
  3) Signed int == int(±ve) — n Bytes
  4) Unsigned int (+ve only) — n Bytes

Topic : Data Types

[NOTE: when there is No specific Nature for data ⇒ Void type]

Float Data Type : Numbers with Decimal Point (or) fractional value (or) Precision

- Memory size in 'c' is 4 Bytes (or) 32-bits

- Type qualifiers : Signed float == float

  long float == double (In 'c', All Real numbers are double type by default)

- If a Real number, need to be treated as float type, then add suffix 'f' to it.

Example:

```
float x = 1.435;
if (x == 1.435)        ⇒ false
    Printf ("GOOD");
else
    Printf ("BAD");
```

o/p: BAD

(1.435 ⇒ double type; x ⇒ float type)

```
float x = 1.435;
if (x == 1.435f)       ⇒ TRUE
    Printf ("GOOD");
else
    Printf ("BAD");
```

o/p: GOOD

(1.435f ⇒ float type; x ⇒ float)

Topic : Data Types

i bits → Signed ⇒ $-2^{(i-1)}$ to $+2^{(i-1)}-1$

→ Unsigned ⇒ 0 to 2^i-1

Data Type	Memory size (bits)	Range of Values
unsigned char	8 bits	0 to 2^8-1 ⇒ 0 to 255
signed char	8 bits	-2^7 to (2^7-1) ⇒ −128 to +127
unsigned int	16 bits (Let)	0 to $(2^{16}-1)$ ⇒ 0 to 65,535
signed int	16 bits	-2^{15} to $(2^{15}-1)$ ⇒ −32768 to 32767
short	8 bits	-2^7 to 2^7-1 ⇒ −128 to 127
long	32 bits	-2^{31} to $2^{31}-1$
float	32 bits	1.7e-38 to 1.7e+38
double	64 bits	3.4e-308 to 3.4e+308
long double	80 bits	-2^{79} to $(2^{79}-1)$ [in Exponential form]

2 mins Summary

- Data Types
  - Primary datatypes
    - Nature
    - Memory size
    - Range of values.

THANK - YOU