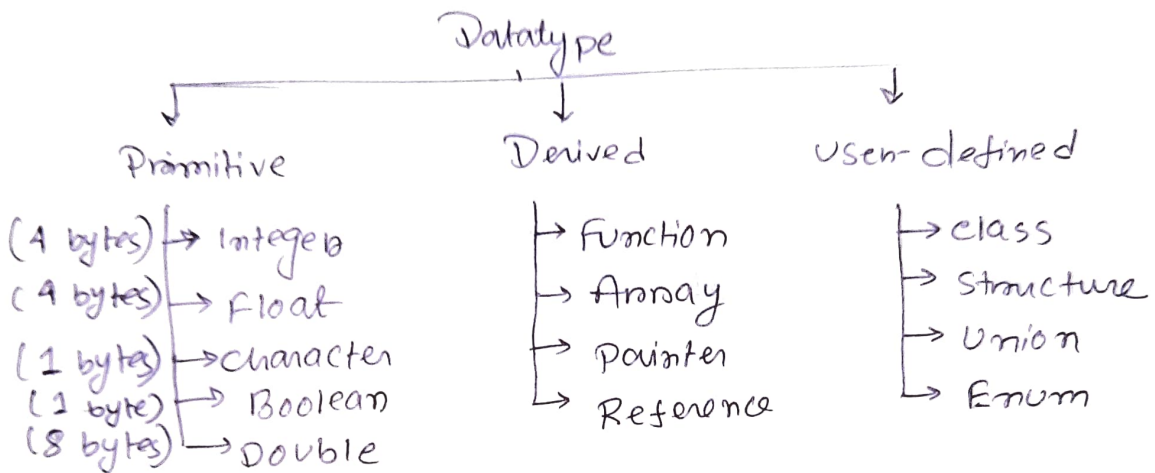


C++ Programming Language

DATATYPE →

variable → varies values anytime in b/w the Program.

Constant → value Constant.



Int^o - Range (unsigned) = 0 to $2^{32}-1$
(Signed) = -2^{31} to $2^{31}-1$

```
#include <iostream>
int main() {
    cout << "Hello World!" << endl;
    return 0;
}
```

Header file include. →

execution of Code begins. →

Output Print

```
int hello;
cin >> hello;
```

Input from user. →

IF-ELSE \rightarrow

```
If (Condition) {  
    statement  
} else {  
    statement  
}
```

```
If (Condition) {  
    statement.  
}  
else if (Condition) {  
    statement;  
}  
else {  
}
```

Nested If-else:-

```
If (Condition) {  
    If (Condition) {  
        statement  
    }  
}
```

Program-1 :- Find max b/w three numbers.
(by using if-else)

Program-2 :- Find even-odd numbers.
($n \% 2 == 0$) \rightarrow even
neither \rightarrow odd.

LOOPS \rightarrow

FOR-LOOP:-

```
for (initialisation ; Condition ; update) {  
    Procedure ;  
}
```

Program-3 :- Sum upto n numbers.

WHILE LOOP:-

```
while (Condition is true) {  
    // body  
}
```

Do-while LOOP:-

```
do {  
    // body  
} while (Condition);
```

(Minimum one execution will happen)

BREAK & Continue:->

Continue → Skip to the next iteration.

break → terminate a loop.

Program-4:- Print all the numbers upto n but which is div by 3 then not print

Program-5:- Find If n is a prime number or not.

Program-6:- Print all the prime number given range a to b.

Prime Numbers:-

Which numbers are only div by 1 & those own numbers.

```
for (int i=2; i<n; i++) {  
    if (n % i == 0) {  
        cout << "Non Prime"    }  
}
```

SWITCH - CASE :-

switch (button)

{

Case 1:

// body
break;

Case 2:

// body
break;

Case 3:

// body
break;

default:

// body;

}

Program - 7 :- Make a Simple Calculator.

OPERATIONS :-

1. Arithmetic operators

→ Binary operators +, -, *, /, %.

→ unary operators ++, --

↓ ↓
Incrementer decreaser.

Pre-Incrementer (++a)

a = 10;
b = ++a;

b = 11

Post-Incrementer (a++)

a = 10
b = a++;

b = 10

2. Relational operators

return bool value.

relation b/w two operands.

== != > < >= <=

Compare
equality

Not
equality

known

3. Logical Operators

AND && Return true if both are true.
OR || Return true if any one true.
NOT ! opposite logical value.

4. Bitwise Operators

AND & $\begin{array}{r} 0101 \\ 0110 \\ \hline 0100 \end{array}$ OR $\begin{array}{r} 0101 \\ 1011 \\ \hline 0111 \end{array}$

^ XOR operators.

$\begin{array}{r} 0101 \\ 0110 \\ \hline 0011 \end{array}$

~ Ones Complement
(0 → 1; 1 → 0)

<< Left Shift operator.

4 << 1
(0100)
= 1000

>> Right Shift operator.

4 >> 1
(0100)
= (0010)

$a \ll n \rightarrow a * 2^n$

$a \gg n \rightarrow a / 2^n$

5. Assignment Operators

=
+= ; -= ; *= ; /=

6. Miscellaneous

sizeof() → return size.

Ternary
operators

Condition ? X ; Y
true
false

'&' → address
of a variable.

'*' → pointer to a value.

Solve Any Pattern : → (Kunal Khoswaha)

How to approach

- ① Run the outer for loop
the no of times you
are having the lines (rows)
Cols
→
↓ rows
- ② Identify for every row number,
* how many col are there
* types of elements in Col.
- ③ what do you need to print.

Task:-

An Kunal Khoswaha
Pattern Question.

Program - 15% → Reverse a given number n .

```
int N;  
int reverse no = 0;  
while (n > 0) {  
    int last digit = n % 10;  
    reverse no = reverse no * 10 + reverse last digit;  
    n = n / 10;  
}
```

Program - 16% → check a Armstrong NO.

Sum of the cube of the digit
is equal to the number.

Functions →

functions is a piece of Code.

```
return-type functionName (Parameters, ....) {  
    function body  
}
```

Program - 17% → Print all Fibonacci Series.

term 1 - 1

term 2 - 2

term 3 - 3 (2+1)

term 4 - 6 (3+2+1)

next term = $t_1 + t_2$;

$t_1 = t_2$;

$t_2 = \text{next term}$;

Program - 18! - Factorial of a number.

$$n! = n \times (n-1) \times (n-2) \times \dots \times 1$$

Program-19:- Calculate nCr & nPr .

$$nCr = \frac{n!}{r!(n-r)!} \quad nPr = \frac{n!}{(n-r)!}$$

Program-20:- Pascal Triangle.

$$\begin{array}{ccccccc} & & & & & & 1 \\ & & & & & 1 & \\ & & & 1 & & 1 & \\ & & 1 & & 2 & & 1 \\ & 1 & & 3 & & 3 & & 1 \\ 1 & & 4 & & 6 & & 4 & & 1 \end{array} \quad \text{term} = {}^nC_r$$

Program-21:- Pythagorean Triplets or Not.

$$x, y, z$$

$$\text{Let } a = \max(x, y, z)$$

then, check property, $a^2 = \text{other sum of the square.}$

Program-22:-

[Binary to Decimal
Octa to Decimal
Hexadecimal to Decimal.
Decimal to Binary
~~Octa to Binary~~
~~Hexadecimal to Binary~~
Decimal to Octa
Decimal to Hexa-decimal

That's Some Problem.