

- Pre / post increment
 - Break / continue
 - Switch.
 - If-loops.
- MCQs (50)

→ Ass / precedence, Binary to Decimal / Decimal to Binary.

Day 4: Doubt clearing session - LB 17/4/2022

⇒ operator precedence

[ek expression ko kaise evaluate karna] →
operator precedence se pata chalega.

Ex:- $\text{int ans} = 2 * \frac{3}{4} + 5$

Refer:

// Operator associativity and precedence chart #

Associativity :- [left to right] ya [right to left]

To avoid confusion → [Brackets use]

Ex:- $\left[\left(\frac{10}{10} \right) * 10 \right]$ $* / \rightarrow \text{same precedence}$
 $= 1 * 10 = 10$

Binary to Decimal / Decimal to Binary

1
101 to 5
 $2^2, 2^1, 2^0$

$4 + 1 = 5$

6 → 110

2 | 6
2 | 3 - 0
1 - 1
110

~~n = 137~~

Kisibi \rightarrow digit ka \rightarrow $\% 10$

Ex:- $n = 137$

$$137 \% 10 \rightarrow 7$$

$$13 \% 10 \rightarrow 3$$

$$1 \% 10 \rightarrow 1$$

$\% \rightarrow$ ans
no \rightarrow chota hojata

loop
untill we reach
all individual
digits

//code:- to obtain, individual digits of
the given number

```
int main() {
```

```
    int n = 137;
```

```
    while(n != 0) {
```

```
        int digit = n % 10;
```

```
        cout << "digit : " << digit << endl;
```

```
        n = n / 10;    // chota kardo n ko [last digit remove  
                        karna hai]
```

```
    }
```

```
    cout << "DONE" << endl;
```

```
}
```

Dry run:- $237 \% 10 \rightarrow 7$

$$23 \% 10 \rightarrow 3$$

$$2 \% 10 \rightarrow 2$$

NOTE:- If we do,

$$x \% N$$

lies b/w $[0 \text{ to } n-1]$

Dry run: $237 \% 10 = 7$

$$\frac{237}{10} = 23$$

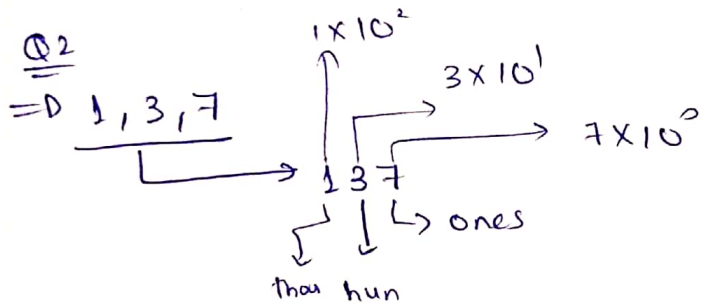
$$23 \% 10 = 3$$

$$\frac{23}{10} = 2$$

$$2 \% 10 = 2$$

$$\frac{2}{10} = 0 \rightarrow \boxed{n=0} \rightarrow [\text{loop rukh jayega}]$$

H/w:- What happens if we take negative number and want to obtain individual digits



H/w:- [code for obtained the number, ~~from~~ by combining individual digits]

Decimal to Binary

If we want to know, it's either 0 or 1

↓
can be done by $[\& 1]$

1	1	1
1	0	0

Dry run: code ke liye

Ex:-

ans $\rightarrow 1. 0 1$

Right shift
 \gg

X X
 $\& 1$
 0

\gg
 Right shift

1

n = zero
 hui shak
 $\gg 0$

//code 2:- Decimal to Binary

```
int main() {
```

```
    int n = 5;
```

```
    while(n != 0) {
```

```
        int bit = n & 1; // bit nikal lo
```

```
        cout << "bit: " << bit << endl; // print kardo,
```

```
        n = n >> 1; // right shift, last bit get destroyed,
```

```
    }
```

o/p:- 1
0
1

o/p for 6:- 1
1
0 $\left\{ \begin{array}{l} \text{ulta aaraha hai ans} \end{array} \right.$

Dry run:- \rightarrow $\begin{array}{c} \text{3rd} \\ 100 \\ \text{1st bit} \\ \text{2nd bit} \\ \text{0} \\ \text{0} \\ \text{1} \end{array}$

$\boxed{\text{ans} = (\text{bit} \times 10^i) + \text{ans}}$

for $i=0 \rightarrow (0 \times 10^0) + 0 = 0 + 0 = 0 \rightarrow 1^{\text{st}} \text{ bit}$

ans = 0

$i=1 \rightarrow (0 \times 10^1) + 0 = 0 \rightarrow 2^{\text{nd}} \text{ bit}$

for $i=2 \rightarrow (1 \times 10^2) + 0 = 100 \rightarrow 3^{\text{rd}} \text{ bit}$

//code3:- Decimal to Binary

[ans o/p me seedha aano ke liye]

```
#include <iostream>
```

```
#include <math.h>
```

```
using namespace std;
```

```
int n=4; //n=7
```

```
int i=0;
```

```
int ans=0;
```

```
while(n!=0){
```

```
    int bit = n & 1;
```

```
    ans = (bit * power(10, i) + ans;
```

```
    n = n >> 1;
```

```
    i++;
```

```
}
```

```
cout << " Binary representation of n is : " << ans  
      << endl;
```

```
}
```

o/p:- 4 → 1
 0
 0

(use karne)

// numb generate kar-vahe hai

~~2081~~
↓
birt

$$n = 4$$

```
2 int bit = n & 1;
```

$\boxed{\text{ans} = (\text{bit} * 10^i) + \text{ans}}$ \rightarrow to generate a number

$$n = n > 1;$$
$$1 + +$$

3.

NOTE:- $n \% 10 \rightarrow$ last digit of no $\leftarrow 123(4)$.
 $n \& 1 \rightarrow$ [last rightmost bit]

$n=6 \rightarrow$

2	6
2	3 - 0
	1 - 1

rem \rightarrow

slpr- 1 1 0 \rightarrow Binary form

lovebabbar3@g

lovebabbar3@gmail.com

$$n=9 \rightarrow \begin{array}{r} 2 \overline{) 9} \\ 2 \overline{) 4-1} \\ 2 \overline{) 2-0} \\ 1-0 \end{array} \uparrow$$

01p = 2001 \rightarrow Binary form.

Binary to Decimal

↓
0 | 1 → 2 bits

$$\begin{array}{ccc} 1 & 0 & 0 \\ \downarrow & \downarrow & \searrow \\ 2^2 & 2^1 & 2^0 \end{array} = 2^2 + 2^1 + 2^0 = 4 + 0 + 0 = 4$$

Nested for:-

```
for(int i=1; i<=n; i++) {
```

```
    for(int j=1; j<=n; j++) {
```

```
        cout << i << " _ " << j << endl;
```

```
    }
```

```
}
```

H/Wi - o/p

Book → Data structures through

C++/C

in depth

Deepali srivastava → pointers

goto

↓

BAD

PRACTICE

Prime no ka flowchart

Let, $n = 13$;

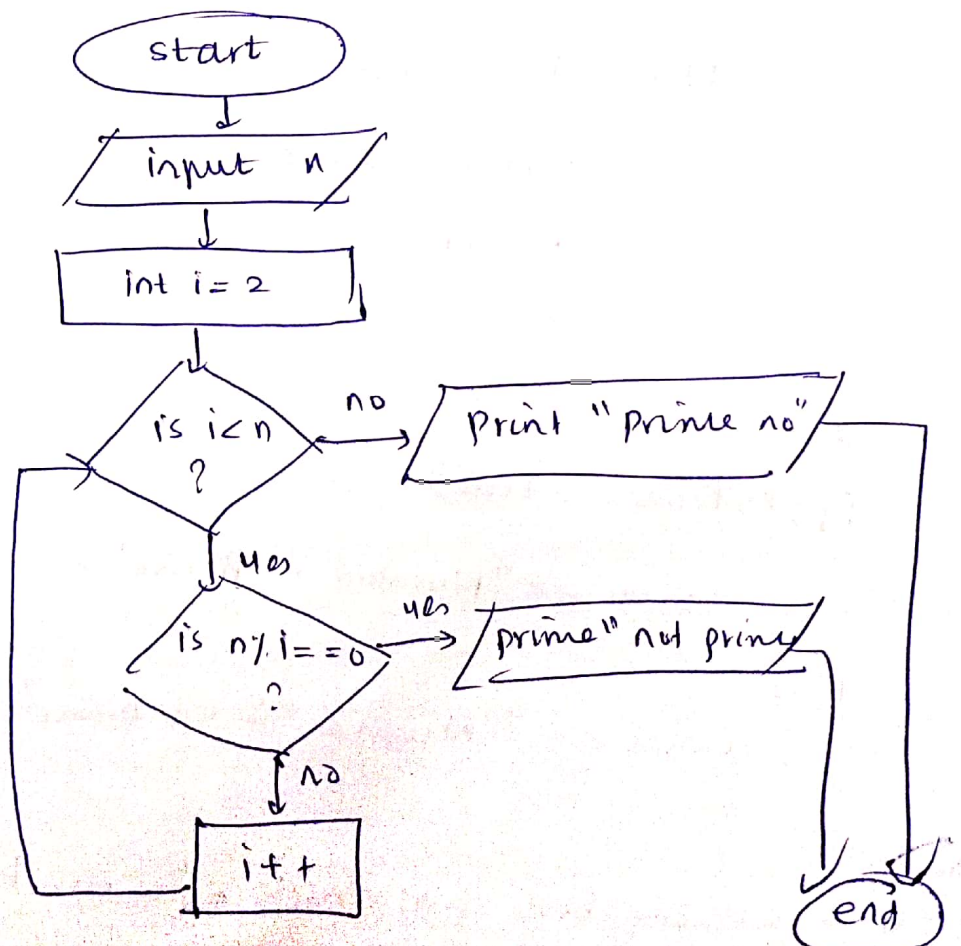
If, $\boxed{2 \text{ to } n-1} \% \rightarrow i \neq 0 \rightarrow \text{prime num}$

$\boxed{2 \text{ to } n-1} \% == 0 \rightarrow \text{not a prime}$

Let, $i = 2$

$n = 13 \rightarrow i < n \xrightarrow{\text{yes}} n \% i = 0 \xrightarrow{\text{yes}} \text{Not a prime}$
Dry run:-
 $\xrightarrow{\text{no}} i++ \rightarrow$

Flowchart for prime no



Code 4:- Enter no to check prime or not

```
int main() {
```

```
    int n;
```

```
    cout << "Enter number to check prime" << endl;
```

```
    cin >> n;
```

```
    bool isPrime = true; // flag
```

```
    for (int i = 2; i < n; i++) {
```

// divide ho gaya kisi number se, it means

not prime

```
        if (n % i == 0) {
```

```
            isPrime = false;
```

```
            break;
```

```
        }
```

```
    }
```

```
    if (isPrime == true)
```

```
        cout << "Number is Prime" << endl;
```

```
    else
```

```
        cout << "Number is not prime" << endl;
```

How

code → without bool → check if a number is prime

or not

$\Rightarrow i \rightarrow \text{creation}$
 $\text{bool} \rightarrow \text{remove}$ } $\rightarrow \text{explore}$

NDA
documentary
↓
Nat Geo

Vishal Dahiyas
↓
Bio

NOTE 1- even $\rightarrow 81 \rightarrow 0$
 $\hookrightarrow \% 2 \rightarrow 0$

odd $\rightarrow 81 \rightarrow 1$
 $\hookrightarrow \% 2 \rightarrow 1$

$\%$ modulus operation \rightarrow heavy operation

$81 \rightarrow \text{last}$

\Rightarrow custom library \rightarrow [Explore] [X \rightarrow not in interviews]

H/W:- B to D } \rightarrow try for -ve number
D to B }

61
26 doubt session