Watch "Vector in ctd"
Before Monday class

# Doubts with Lakshay Bhaiya [Week 3]

Special class

Love Babbar • Sept 10, 2023

Type casting

Switch case

# Functions

Special class

float x = 4.0; why x=4?

int arr[5] = { }

char x x

char arr2[5] = arr2 = ( )arr

Love Babbar • Sept 4, 2023

```
int arr[5] = {1, 2, 3, 4, 5}

char arr2[5];

arr2 = arr;

arr2 = (char)(arr);
```

arr ??

arr[1]

=>

int arr[5]; = "Raw allocation"

65.0

Vector

char -> 65 A

Char a = 'A';

65

a

Tarun →

```
float x = 5.0;
```

```
cout<<x;
```

→ 5

5.0

Cout → 5

```
int a = 3.2;
```

↳ error??

Functions :- a block of code / sub-program → function

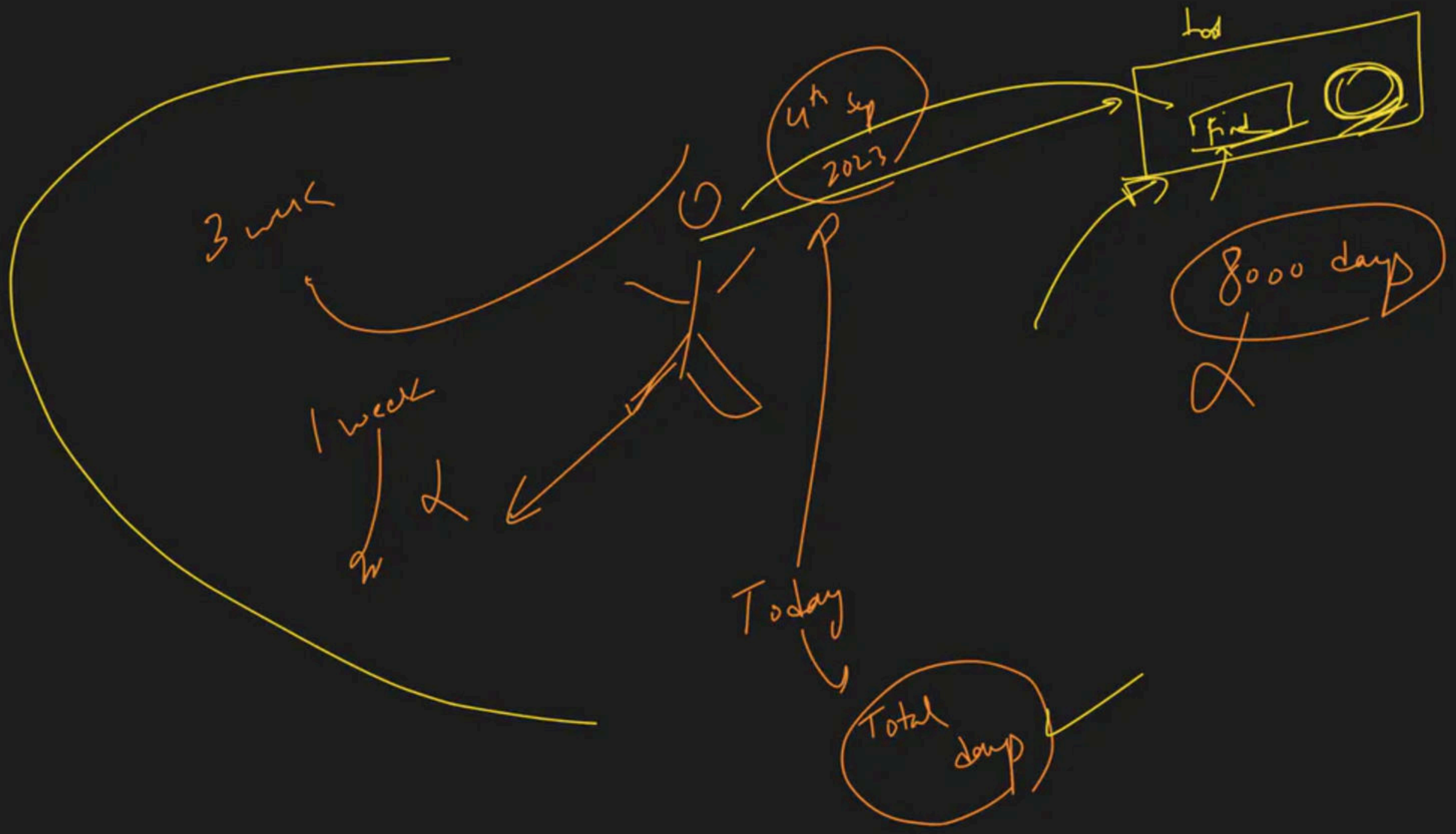
that is linked to a well-designed task

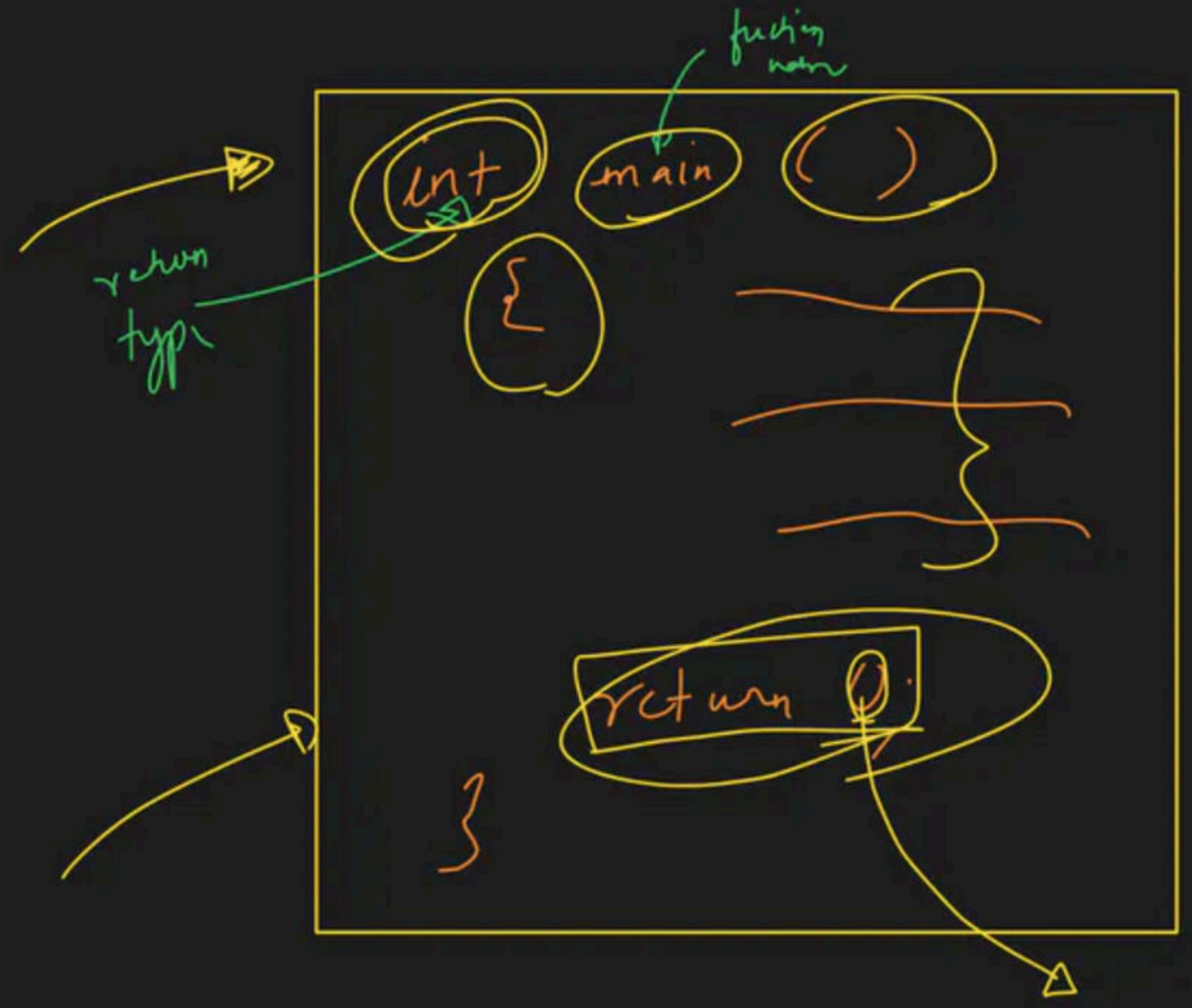
why → Reusability, Readable

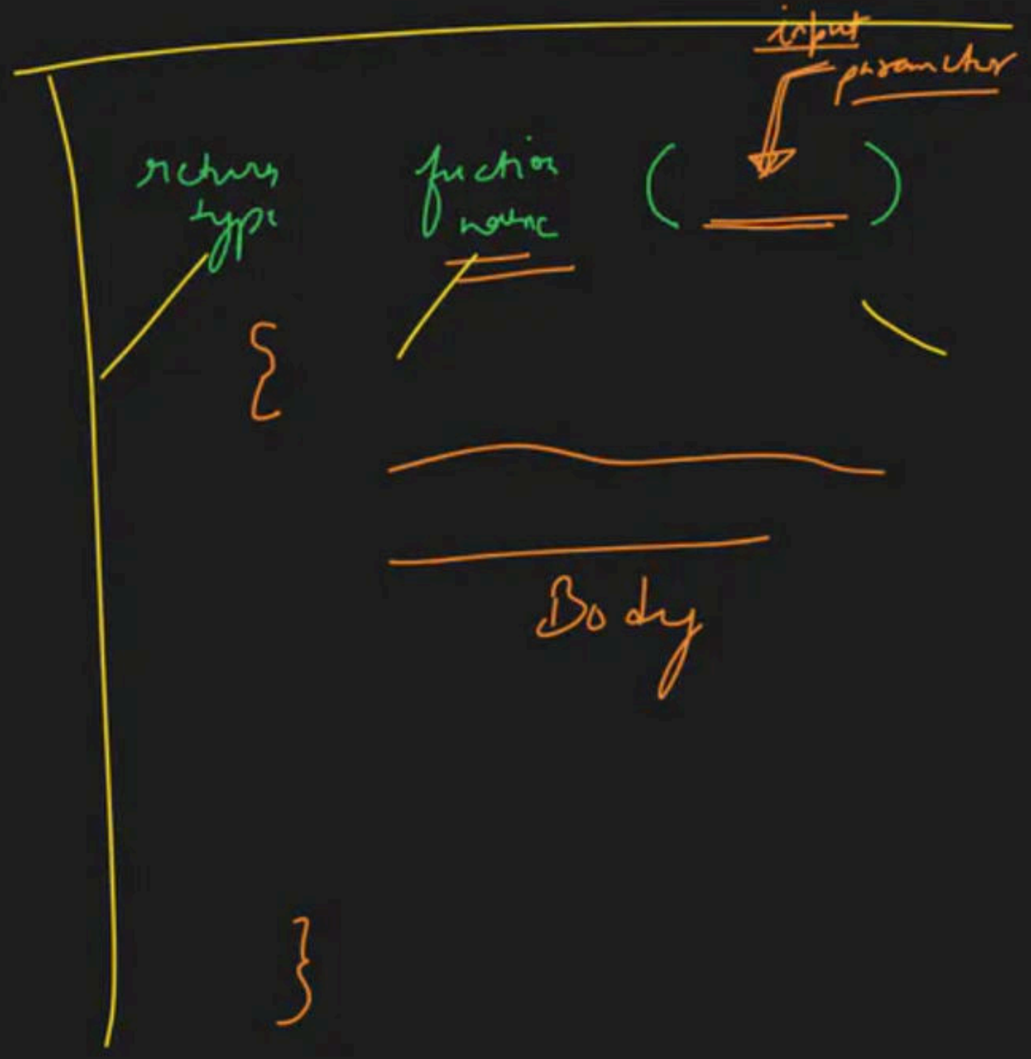
Logic / Algorithm → func

Issue

- Lengthy / Bulky
- Buggy
- Readability
- Reuse


3 week
4th Sep 2023
Find
8000 days
1 week
Today
Total days


function name
int
main
( )
return type
{
return 0;
}


input parameter
return type
function name
( )
{
Body
}

Simple function

print 10 times

"Sundar Ko Sundar pasand hai"

(1)

```
void printLine (      )
{
    for (i=0; i<10; i++)
    {
        cout << "Sundar Ko ———————— ";
    }
}
```

```
int main()
{
    printLine();
    return 0;
}
```

function call

```
void printLine()
{
    for (i=0; i<10; i+1)
    {
        cout << "_____";
    }
}
```

add 2 numbers

fn

# Function

declare

```
void printMessage ();
```

```
int addNumbers (int a, int b);
```

define

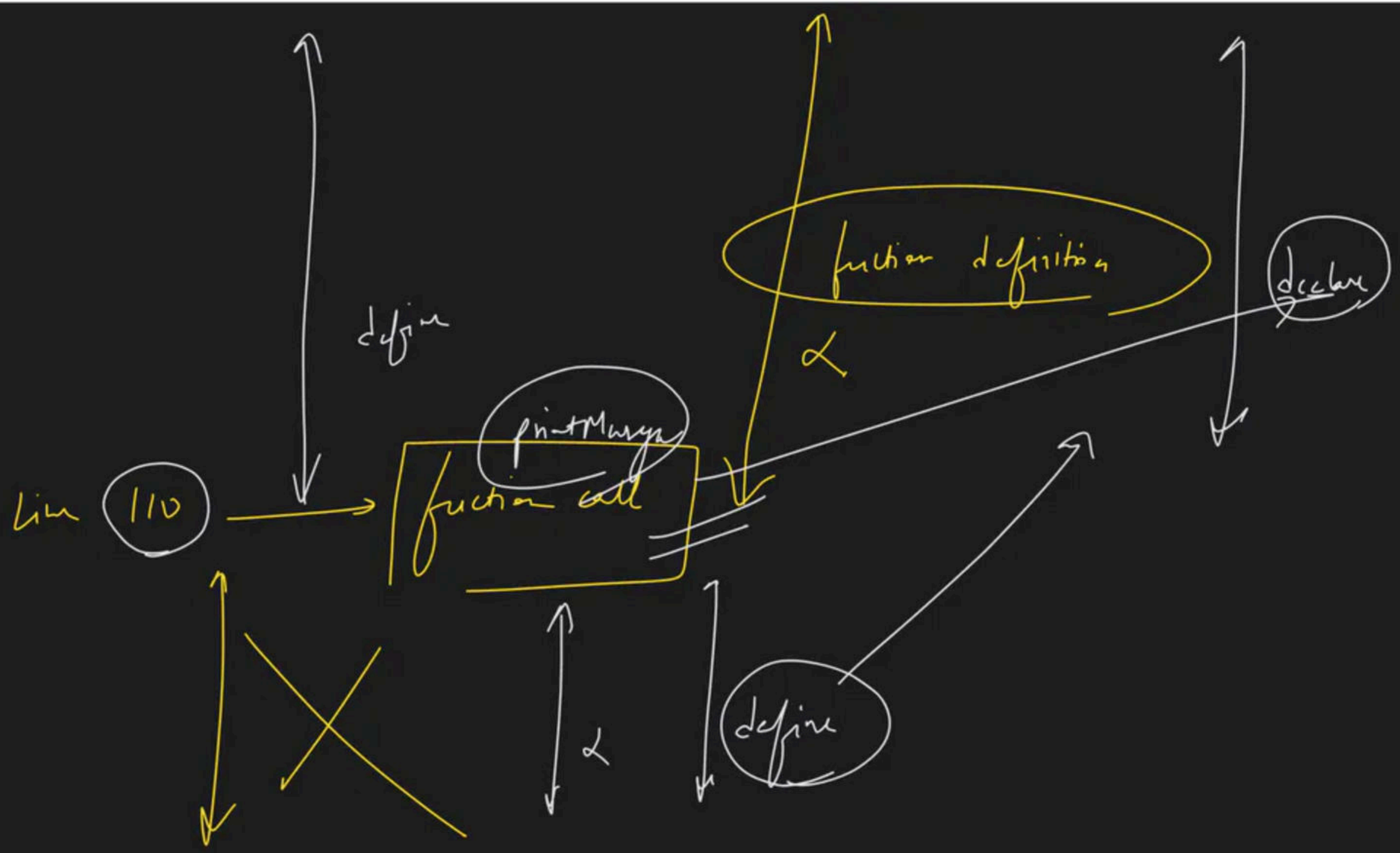
```
int addNumbers (int a, int b)
{
    int sum = a + b;
    return sum
}
```

Function call

```
printMessage();
```

```
function Name ();
```

```
addNumber (5, 3);
```


define
function definition
declare
α
printMargin
function call
line 110
α
define

functions

(A)

return

int
char
array
string
bool

(B)

return

void

① write a program that take n1 & n2 as i/p & returns their sum as output

② write a pg to print the sum of n1 & n2

Entry Point

```
int main ()
{
    cout << "Hi";
    printA();
}
```

```
printA ()
{
    cout << "I am inside A"
}
```

→ function return hogya → wapas jao

1 min

output → Hi
I am inside A

# Function Call Stack :-

- function call
- local variable
- Kr fun → Ko fun
  Ko call Kr
- return value

Stack → ?

Last - in first Out

LIFO

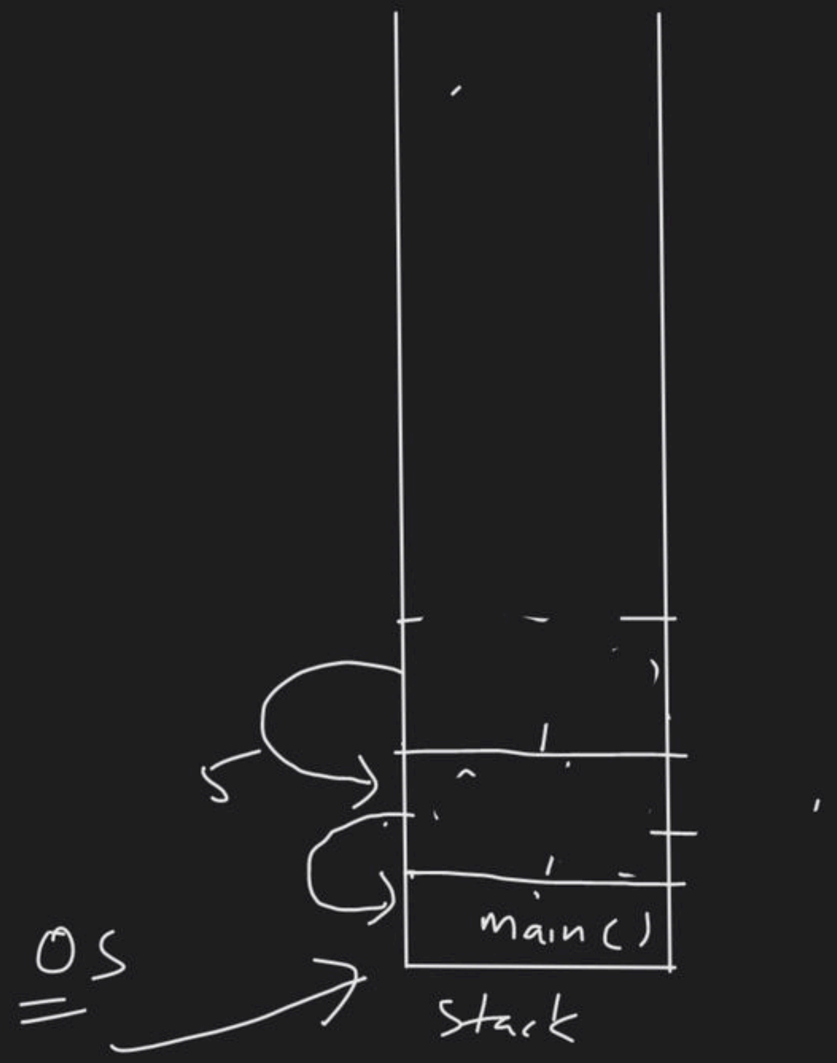
Discord → ?

```
int fun2( )
{
  cout 5;  return 5;
}

int fun1( int a)
{
  cout<<a
int c= fun2(a); return c;
}

int main ( )
{
cout<<fun1(10);

}
```

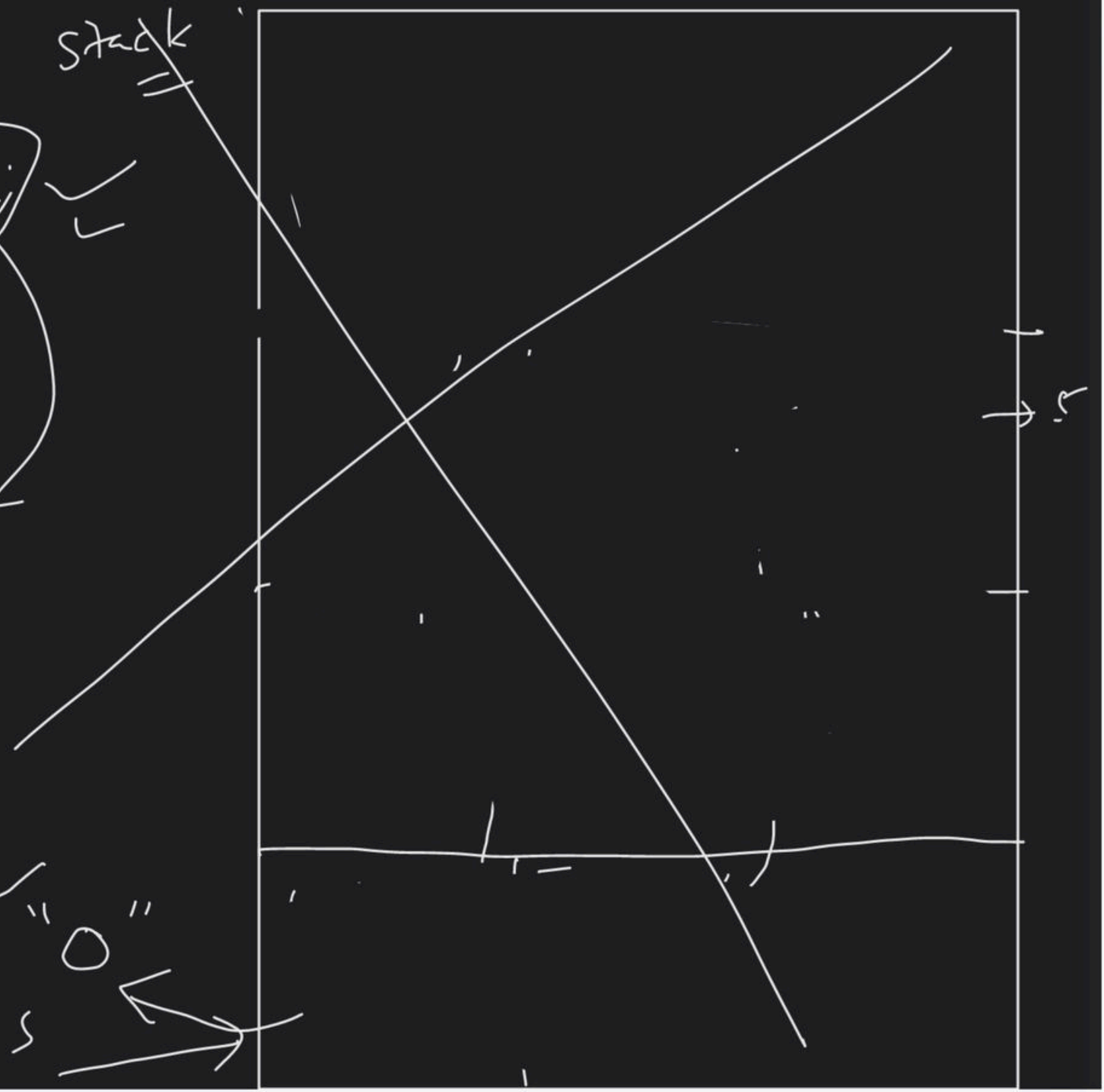


```
int fun2(int a)
{ cout << a ; return a;
}

int fun1(int a)
{
    int c = fun2(a)
    return c;
}

int main()
{ int a = fun1(5);
  return 0;
}
```

stack




"0"

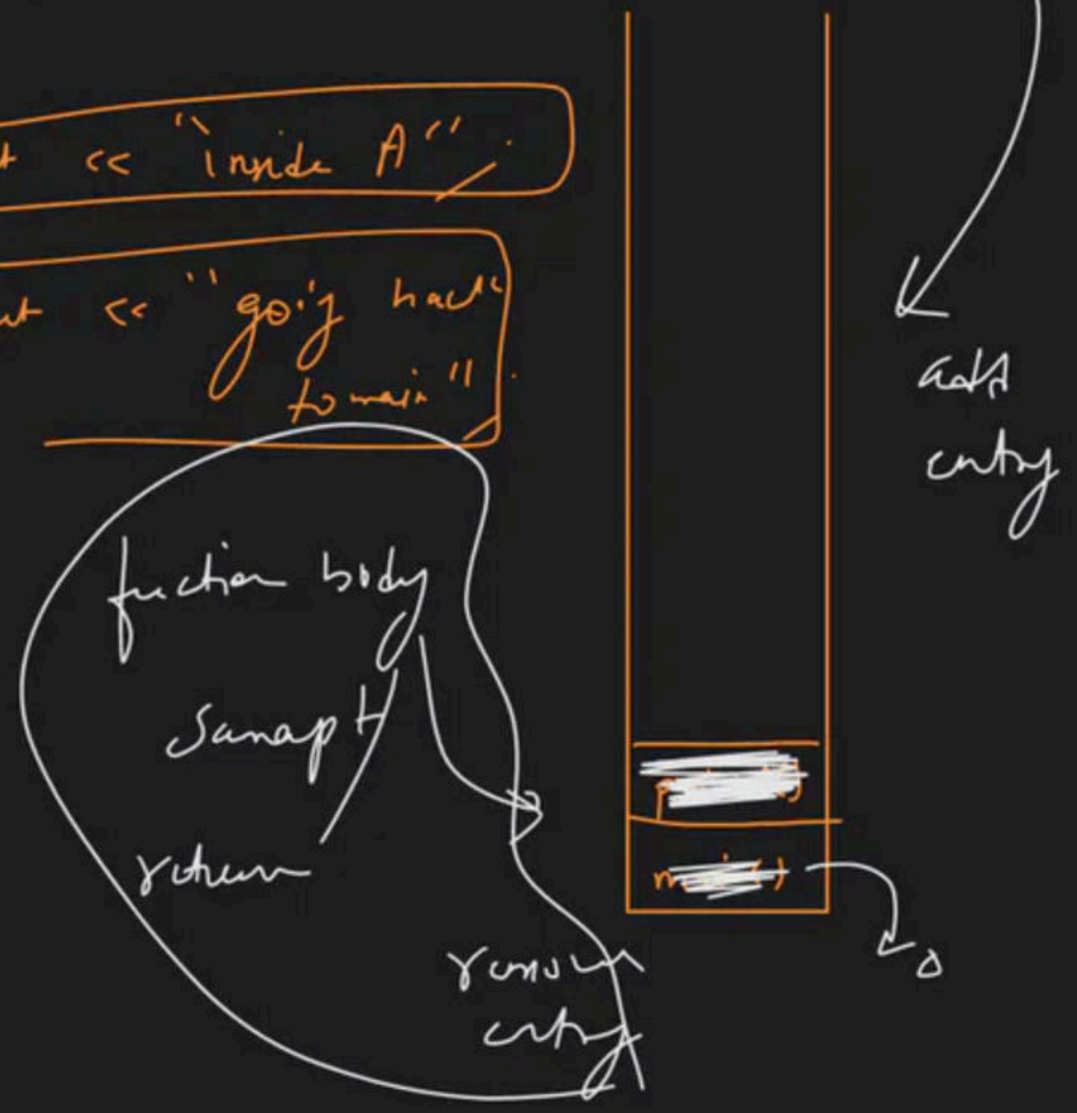
OS

```
int main ()
{
    cout << "inside main";
    printA ();
    cout << "Back in main";
    return 0;
}
```

```
void printA ()
{
    cout << "inside A";
    cout << "going back to main";
}
```

o/p → inside main
inside A
going back to main
back in main

function call



```
int main()
{
    printA();
}
```

```
printA()
{
    cout << "Inside A"
    printB();
}
```

```
printB()
{
    cout << "Inside B"
    print()
}
```

```
print()
{
    cout << "Inside C"
}
```

Q1 → write a function to print sum of 3 no.

i/p → 3 no

o/p → print sum

→ void

→ write a function to return sum of 3 no.

→ int

```
int add (int a, int b, int c)
{
    int sum = a + b + c;
    return sum;
}
```

4 + 4 + 5

13

sum

```
int main ()
{
    int ans = print_sum(4, 4, 5);
```

13

13

ans

```
int main()
{
    messagePrint();
}
```

```
messagePrint()
{
    cout << "msg 1"
    return;
    cout << "msg 2";
}
```

2 min pacni

msg 1

```
int main()
{
    return 0;
}
```

→ find max of 3 no'

a, b, c

① find max of 3 no a, b, c → Pre-defined

② Counting from 1 to N

③ Check prime no or Not

④ Check Even or Odd

⑤ Sum of all No upto 1 → N

⑥ Sum of all Even no upto 1 → N

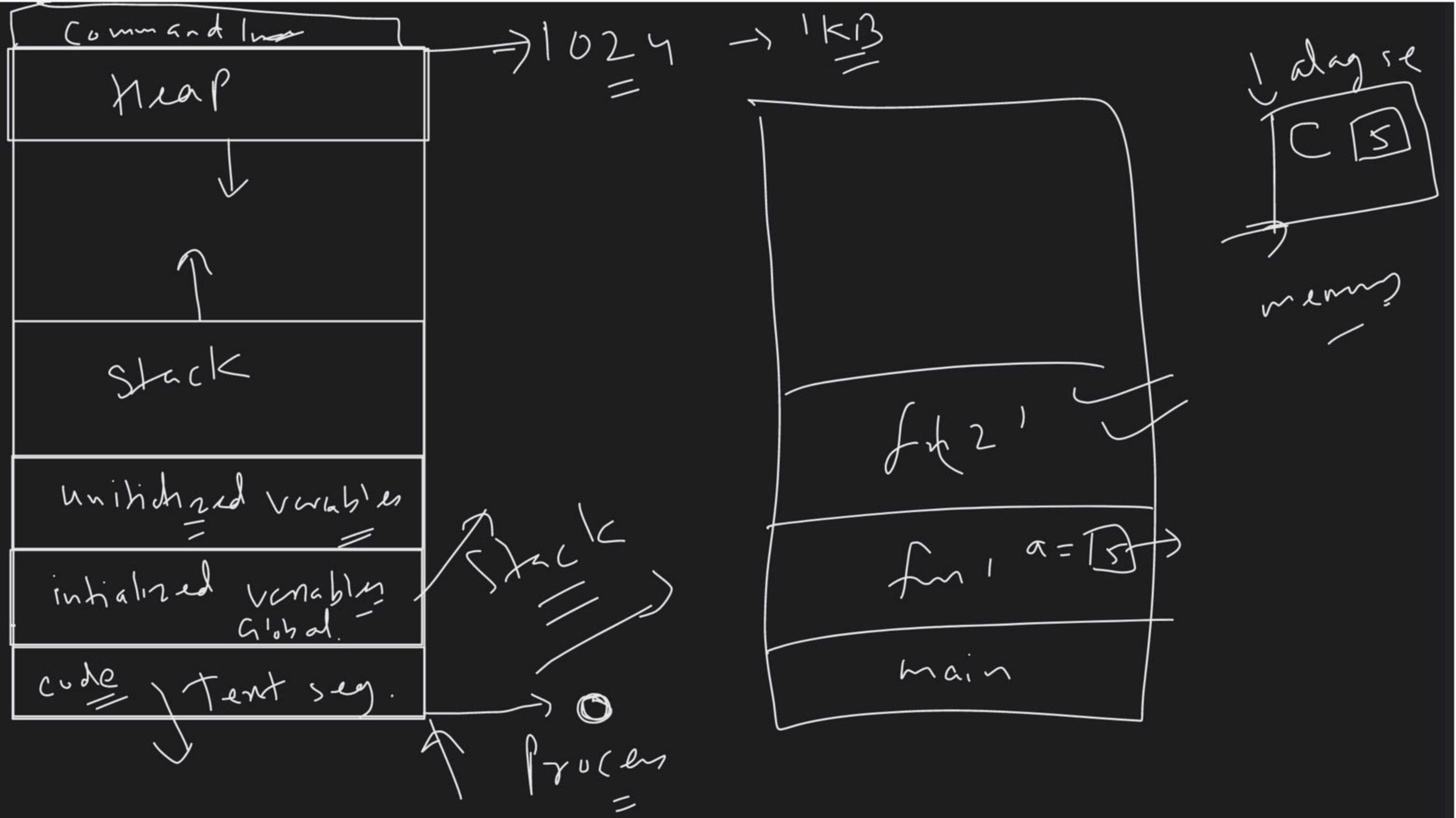
→ class Work

pre-defined
inbuilt

function

max(x, y)

r


Command line
Heap
Stack
uninitialized variables
initialized variables
Global.
code
Text seg.
1024
1KB
Stack
Process
fn2
fn1
a = 5
main
1 dangle
C
5
memory

$\Rightarrow$ Global

for (

{

}

sum = 0

1, 2, 3, — — N

0+1+2+3 — — +N

0+1+2+3 — — +N

```
void findSum(int n)
{
    int sum = 0;
    for (i=1 ⟶ <=10; i++) ⟶
    {
        sum = sum + i
    }
    cout << sum;
}
```

sum = sum - i

sum -= i

55

$0+1+2+3+4+5+6+7+8+9+10$

$=55$

sum

i=1

i=2

i=3

i=4

i=5

i=6

i=7

i=8

i=9

i=10

55

sum = sum + i

or

sum += i

Max → a, b, c

max(n1, n2)

a, b,

max

ans

c

max(max(a, b), c)

max

# Home Work

- Function to find area of circle
- Function to find factorial of a No.
- print all prime no from 1 → N
- print all digits of an Integer → Ultra imp
- create a no using digits → Ultra imp
- print Binary rep of a decimal no → Ultra imp

- Convert Km into miles
- convert Farhenheit into Celcius
- Count all set bits of a Number
- check Even/odd using bitwise operator

1 min

loop → so we the

for ( i=0 i<n)

i = 0, 1, 2, 3 - - - - n-1

for ( i=0; i<=n) i = 0, 1, 2, 3 — n

for ( i=1; i<=n) i = 1, 2, 3 - - - - n

for ( i=1; i<n) → i = 1, 2, 3 - — — n-1

# Check prime or Not

i/p → 7

N

1 N

2, 3, 7, — — N − 1

perfectly divisible

% → 0

prime no → N

1 divisible

self

size of array

# Array Level-2 [Join Here]

Special class

**Love Babbar** • Sept 8, 2023

```
int main()
{
    int a=5;
    solve(a);
    cout << a << endl;
}
```

104

5

a

```
void solve(int a)
{
    a++;
    cout << a << endl;
}
```

204

~~5~~ 6

a

Passing Value

hum ek copy create hoti

o/p

→ 6

→ 5

```
int main()
{
    int mark = 90;
    mark++;
    solve(mark)
    cout << mark << endl;
}
```

o/p →

90 ~~91~~ ~~91~~ 100

marks, m

```
solve ( int m )
{
    m--;
    m = m * 10;
    cout << m;
}
```

→ 90

100, 900

```
int main()
{
    int sundari = 100;
    sundari--;
    sundari -= 5;
    solve(sundari);
    cout << sundari;
}
```

sundari = sundari - 5
= 99 - 5
= 94

~~100~~ ~~99~~ ~~94~~ 93

sundari gaadu

o/p

93

```
void solve(int& gaadu)
{
    gaadu--;
    cout << gaadu + 10;
    return;
}
```

93 + 10 → 103

103 93

```
int main ()
{
    int a = 5;
    solve(a);
    cout << a;
}
```

```
void solve (int & a)
{
    a++;
    cout << a << endl;
}
```

104 ~~5~~ 6

a = 4

6

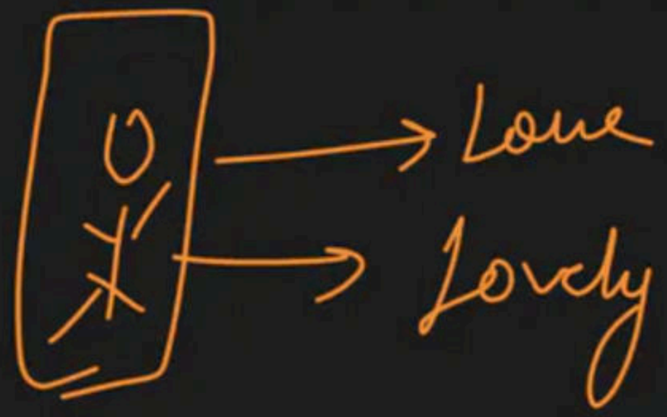
6

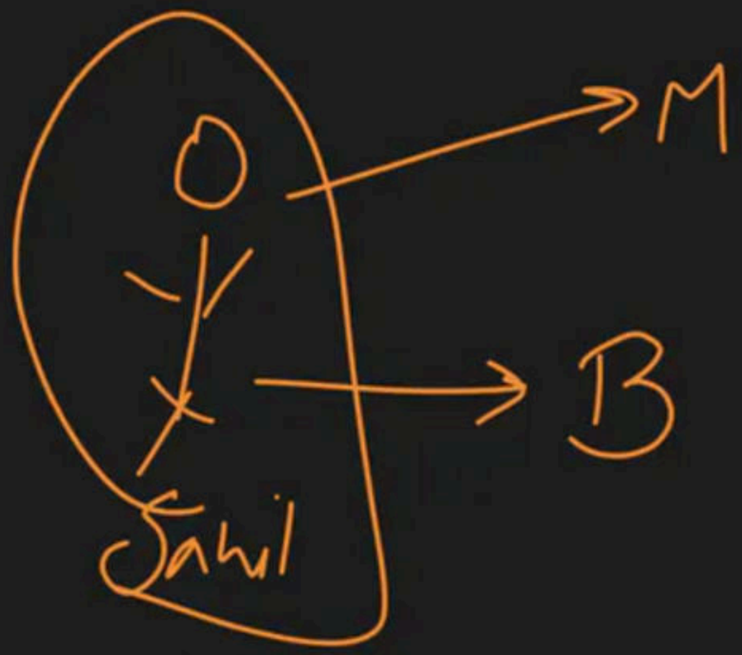
6

6, 6

love

lovely




Love

Lovely

int a = 5

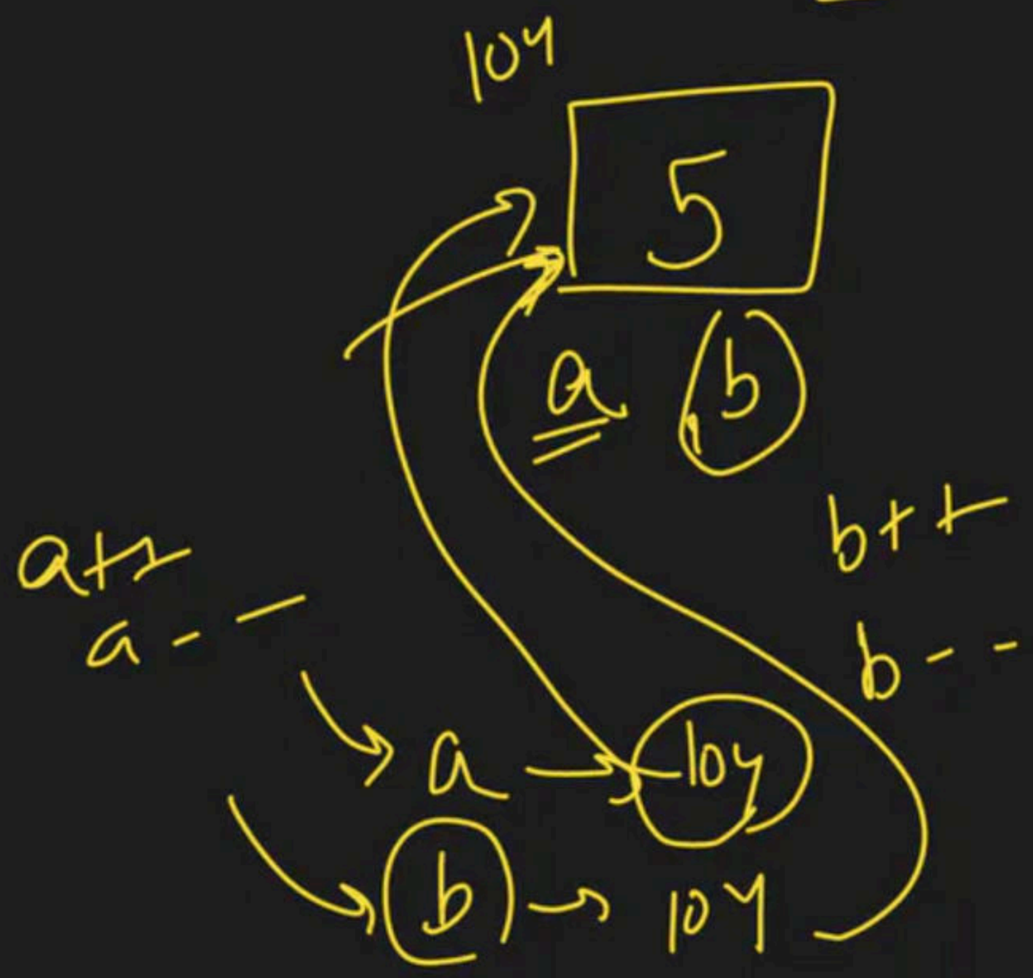
int & b = a

ref. variable

same mem location

but

different names



```
int main()
{
    int arr[] = {1, 2, 4}
    int n = 3
    solve(arr, n)

    for (int i=0; i<n; i++)
    {
        cout << arr[i]
    }
}
```

arr

| | | |
|---|---|---|
| 1 | 2 | 4 |

```
solve(int arr[], int size)
{
    arr[0] = 100;
}
```

| | | |
|---|---|---|
| 1 | 2 | 4 |

pass by V

| | | |
|---|---|---|
| 100 | 2 | 4 |

pass by ref

```
int main()
{
    int n = 3;
    int arr[] = {10, 20, 30}
    solve(arr, n);

    // print array
    for (i=0; i<n; i++)
    {
        cout << arr[i] << " ";
    }
}
```

n: 3

arr: 100 | 20 | 30 (0, 1, 2)

```
void solve (int arr[], int n)
{
    arr[0] = 100;
}
```

copy of this begin

copy begin

arr: 10 | 20 | 30

n: 3

n: 3

→ find Unique Element

↳ i/p → 2, 10, 11, 13, 10, 2, 13, 13, 13

ans

each element → occurs twice except one → find out

U.E → 11

How → How to solve ⇒ XOR → same val → 0

→ diff val → 1

$1 \wedge 0 \wedge 0 \wedge 0$

$0$

$0$

$1$

$0 \wedge a \to a$

$0 \wedge z \to z$

XOR

| a | b | o/p |
|---|---|---|
| 0 | 0 | 0 |
| 0 | 1 | 1 |
| 1 | 0 | 1 |
| 1 | 1 | 0 |

Code → Optimal ✗ But ✓ sol^n

→ Ex solution

→ Brute form → Basic sol^n

$\boxed{9}$

n

arr

| 10 | 2 | 0 ~~11~~ | 10 | 2 | 13 | 15 | 13 | 15 |
|---|---|---|---|---|---|---|---|---|
| 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 |

$$ans = 0 \wedge \cancel{10} \wedge \cancel{2} \wedge \cancel{11} \wedge \cancel{10} \wedge \cancel{2} \wedge \cancel{13} \wedge \cancel{15} \wedge \cancel{13} \wedge \cancel{15}$$

$$= 0 \wedge 0$$

$$ans = 0$$

i/p → array → [ ] → {10, 20, 30}

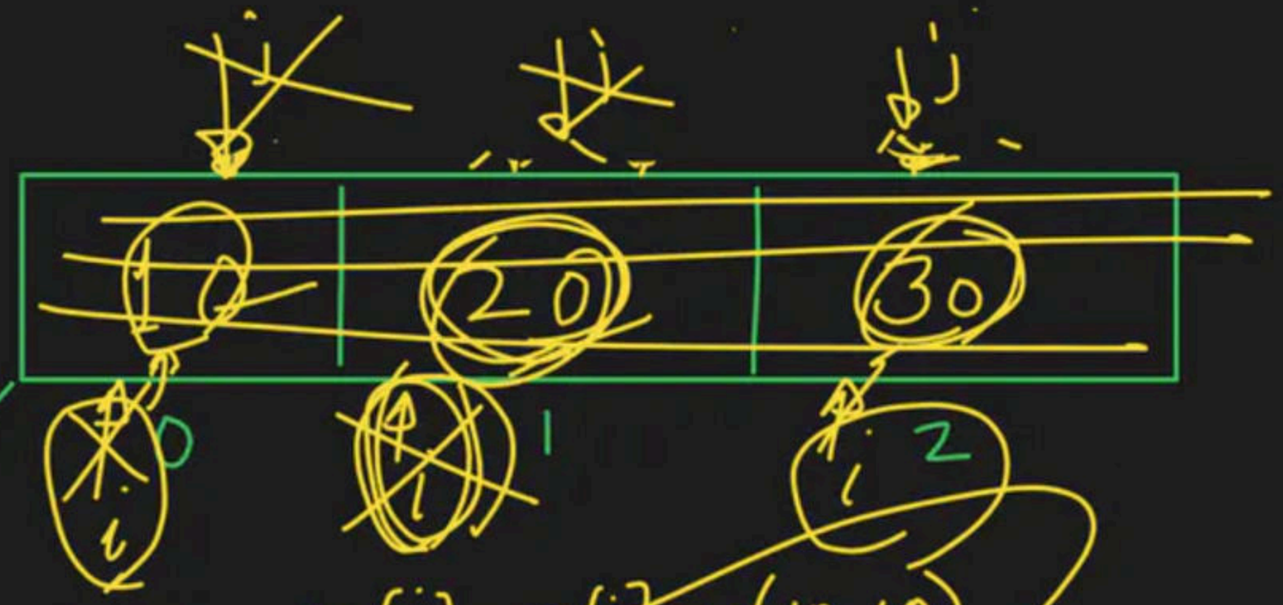
Print all pairs

2 min

| | | |
|---|---|---|
| (10, 10) | (20, 10) | (30, 10) |
| (10, 20) | (20, 20) | (30, 20) |
| (10, 30) | (20, 30) | (30, 30) |

2 loop

H/W — Pair Sum / Two Sum




row 0 →
row 1 →
row 2 →

nested loop

$i=0$, $j=0$ → arr[i], arr[j] → (10, 10)

$j=1$ → (10, 20)

$j=2$ → (10, 30)

$i=1$, $j=0$ → (20, 10)

$j=1$ → (20, 20)

$j=2$ → (20, 30)

$i=2$, $j=0$ (30, 10)

$j=1$ (30, 20)

$j=2$ (30, 30)

```
for ( i = 0; i < n/2; i++)
{
    for ( j = n-i; j >= 0; j--)
    {
        cout << i << " " << j;
        cout << arr[i] << " " << arr[j];
    }
}
```

odd %

```
for (int i=0; i<n; i++)
{
    for (int j=0; j<n; j++)
    {
        cout<< i << j ;
    }
}
```

$i=0;\ j=0; \rightarrow (0,0)$

$i=0\ \ j=1 \rightarrow (0,1)$

$i=0\ \ j=2 \rightarrow (0,2)$

$j=3$

$i=1\ \ j=0 \rightarrow (1,0)$

$j=1 \rightarrow (1,1)$

$j=2 \rightarrow (1,2)$

$i=2\ \ j=0 \rightarrow (2,0)$

$j=1 \rightarrow (2,1)$

$j=2 \rightarrow (2,2)$

$i=3 \rightarrow f \rightarrow \alpha$

i/p → array → [1, 2, 3, 4]

→ print all triplets

```
for (i=0; i<n; i++)
{
    for (j=0; j<n; j++)
    {
        for (k=0; k<n; k++)
        {
            cout << arr[i] << arr[j] << arr[k];
        }
    }
}
```

H/w → Three Sum / Triplet Sum

→ Sort 0's & 1's

n = 9

I/P →

| 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 |
|---|---|---|---|---|---|---|---|---|
| 0 | 0 | 0 | 0 | 0 | 0 | 1 | 1 | 1 |

all

o/p → 0 0 0 0 0 0 1 1 1

zero → 6

one → 3

① Counting → Kyke dikhata hai

② 2 pointer approach → H/W

③ sort ( ) → increasing

Logic → count → 0 & 1

→ place 0

→ place 1

$\Rightarrow N = 5 \rightarrow 000101$

$\Rightarrow$

```
while (n)
{
    bit = N & 1;
    if(bit)
        count ++;
    N = N >> 1;
}
```

Bitwise

$\Rightarrow$

$\Rightarrow$ 5 & 1 $\rightarrow$

$$\begin{array}{r} 000101 \\ \& \quad 000001 \\ \hline 000001 \end{array}$$

N & 1 $\rightarrow$ LSB

C = 01

⇒ N = 5 →

101

001

001

5 & 1 = 1

000010

000010

N = 5

1

bit = N & 1 = 1

if (Bit) → count++

N = N >> 1;

```
count = 0
while (n)
{
    n = n & (n-1);
    count++;
}
```

3

count set Bit

$N = 5$

8 4 2 1

$C = 0$

① $n = 5 \& (n)$

$$\begin{array}{r} 101 \\ \& \; 100 \\ \hline 100 \end{array}$$

② $N = 4$

$N = 4 \& 3 \rightarrow$

$$\begin{array}{r} 100 \\ 011 \\ \hline 000 \end{array}$$

$N = 0$

8421

$N = 15$

while (n)

$n = n \& (n-1)$

CTT

(1) $N = 15 \& (15-1) = 15 \& 14$

$$\begin{array}{r} 1111 \\ \& \ 1110 \\ \hline \rightarrow 1110 \end{array}$$

(2) $N = 14$

$N = 14 \& (13)$

$$\begin{array}{r} 1101 \\ \hline 12 \rightarrow 1100 \end{array}$$

(3) $N = 12$

$N = 12 \& (11)$

$$\begin{array}{r} 1011 \\ \hline 8 \leftarrow 1000 \end{array}$$

(4) $N = 8$

$N = 8 \& 7$

$$\begin{array}{r} 0111 \\ \hline 0 \ 0000 \end{array}$$

$N = 0$

count = 3

```
while (count --)
{

}
```

0

```
while(3)
{

}
```

```
while(2)
{

}
```

```
while(1)
{

}
```

```
while(0)
{

}
```

```
while (marks - -)
{

}
```

total kitni baar

or

total kitni iteration

marks

```
while ( zero count -- )
{

}
```

→ 6 times

```
while (one count --)
{

}
```

→ 3 times

Step 1 counting

```
int zerocount = 0;
int onecount = 0;

for( i=0; i<n; i++)
{
    if (arr[i] = = 0)
    {
        zerocount++;
    }
    if (arr(i) - - = 1)
    {
        onecount++;
    }
}
```

zerocount = 6

onecount = 3

Step B

```
int index = 0

while (zerocount --)
{
    arr[index] = 0;
    index++;
}
```

6 times

Step 3

```
while( onclout --)
{
    arr[index] = 1;
    index ++;
}
```

Shift arrays element by 1

K element

60

top

i/p

| | | | | | |
|---|---|---|---|---|---|
| 10 | 20 | 30 | 40 | 50 | 60 |
| 0 | 1 | 2 | 3 | 4 | 5 |

o/p

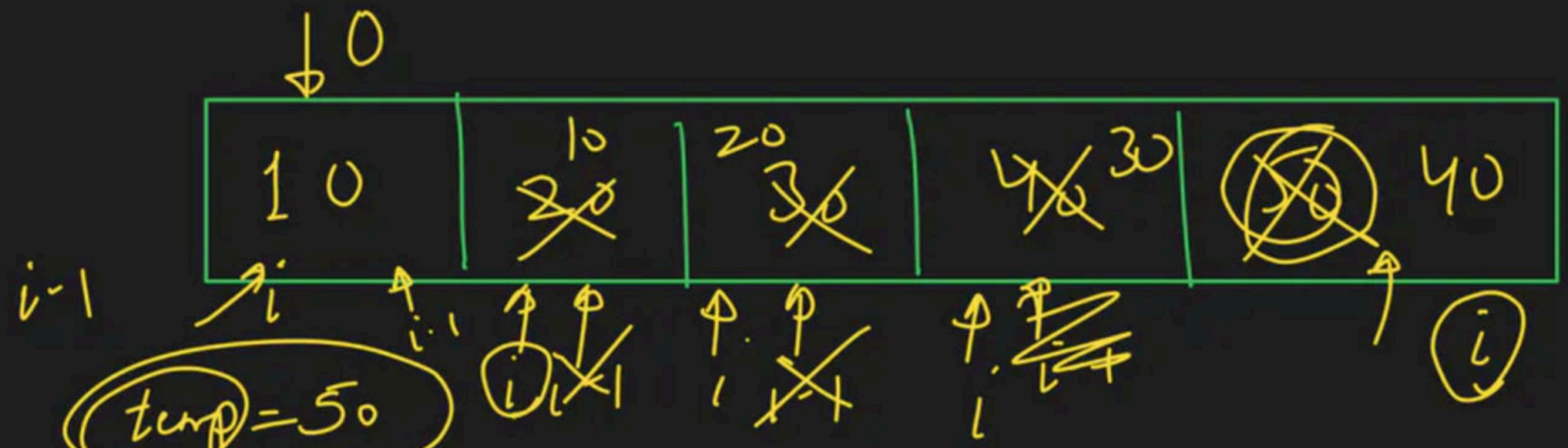
| | | | | | |
|---|---|---|---|---|---|
| 60 | 10 | 20 | 30 | 40 | 50 |
| 0 | 1 | 2 | 3 | 4 | 5 |

i-1

i

$\cancel{arr[i+1] = arr}$

$arr[i] = arr[i-1]$




(A) temp = 50

(B) i = n - 1 → ~~i = 0~~ i = 1

→ arr[i] = arr[i-]

(C) arr[0] = temp

K = 2

| | | | | | | |
|---|---|---|---|---|---|---|
| 50 ~~10~~ | 60 ~~20~~ | ~~30~~ 10 | ~~40~~ 20 | ~~50~~ 30 | ~~60~~ | 40 |

① temp[] = {50, 60}

② shift

③ copy temp

30 40 50

⇒ 0 | 1 | 0 | 1 | 0 | 0 | 1 | 1 | 1 | 0

| 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 |
|---|---|---|---|---|---|---|---|---|---|
| 0 | 0 | 0 | 0 | 0 | 1 | 1 | 1 | 1 | 1 |

l

h

h

int l = 0

int h = n-1

⇒ 10-1 = 9

① increment 'l' Jab Jab hme '0' dikhega

② decrement Jab Jab 1 dikhe

③ swap(a[l], a[h])

l++;

h--;

| 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 |
|---|---|---|---|---|---|---|---|---|---|
| 0 | 1 | 0 | 1 | 0 | 1 | 1 | 1 | 1 | 1 |

l

m

h

```
if (arr[l] == 0)
    l++

else if (arr[h] == 1)
    h--;

else {
    swap → l++
           h--;
}
```

```
while ( l < h )
{

}
```