

- Why C language is imp?
- Onddele is written in C
- Core libraries of android are written in C
- MySQL is written in C
- Almost every device driver is written in C

Alag alag hardware ko chalane ke liye alag alag programming ki jati hai unko device driver kehte hai

- Major part of web browser is written in C
- Unix operating system is developed in C
- C is the world's most popular programming language

- For students

- C is imp to build programming skills
- C covers basic features of all programming languages
- Campus recruitment process
- C is most popular language for hardware dependent programming

### History of C language

Maxim Richards BCPL = Basic Combined programming language

Ken Thompson BCPL improve  
Kauke B language banai 1969

Dennis  
Ritchie

NUC 1078

(39)

C language  
UNIX operating system.

## Fundamental Terminologies

- Comp is an electronic device that takes input, process it and gives output.

electronic = Current ni lagega usme current ki supply kam hoti hai

electric = Tatkha lagega current ki supply jayada hoti hai

- What are 0 and 1?

Ans Coding Duniya ki hau ek cheez ko 0 ya 1 ki form me code kame ka tareka nikal rakha hai.

Hard disk = Usme read write jo head hota hai wo magnetic material ke andar information store karta hai aur N pole aur S pole banata jata hai. Agar N pole niche banata hai to 1, N pole upper banata hai to 0 bolta hai. Magnetic material me kai saari places hoti hai uss puri hard disk me apna bahut sare 0 aur 1 gare hote hai

*16/8/2024*  
RAM = Capacitors me charge hold hota hai. Agar capacitors me charge hold hota hai to 1. Agar charge hold ni hota to 0 ek 0, 1 ko same kaeme ke liye ek capacitor

haud  
nai  
ho.

disk me info hamesha ke liye store ho sakti  
chahiye usme change ho ya na

PAGE NO.

DATE: / /

RAM me data change ki form me store nota hai isliye  
vo volatile memory hai

- Processor = me chote chote registers note hai jisme  
hum info save karte hai. flip flop  
ki madad se. Flip flop kya hai vo ek  
byte ka store karne ka wala circuit  
hai. 16 flip flop ko milakar ek device  
banainge to banega 16 flip flop a  
ek register. 16 o, off

Voltale

- Hardware -

Monitor, keyboard, scanner, etc. sab Hardware  
nota hai. Ye ek physical chey hoti hai jiske  
touch kar sakte ho.

- Data file ke roop me comp ke andar nota hai.  
File is a data bundle

file name → file extension → file kis format ki hai usne kit  
track1.mp3 → se pata jata hei taaki ka data rakh  
ka

jpg form me hai to isse image ki pixels ki  
coding mani jati hai

txt = koi character ya koi symbol.

software exe = executable file. Special data iske  
o, i ka matlab comp ke liye midesh

ki comp ko kaerna kya hai  
exe file software.

PAGE NO.	5
DATE:	1 / 1

Application software - User ke matlab ke note  
hai: Calculation. Jisse user ko " " hota hai

System software = Jo machine ke matlab  
ke note hai . Eg graphics  
System ke liye karam karne re hai ki  
System sahi se chale.

Program and Process.

Planning  
Set of instructions is c/d program

Active state of program is c/d process  
jab kisi program ko  
run karta hu execute karta hu tab

Operating System = It is a system software  
matlab ye hardware se  
interact karta hai

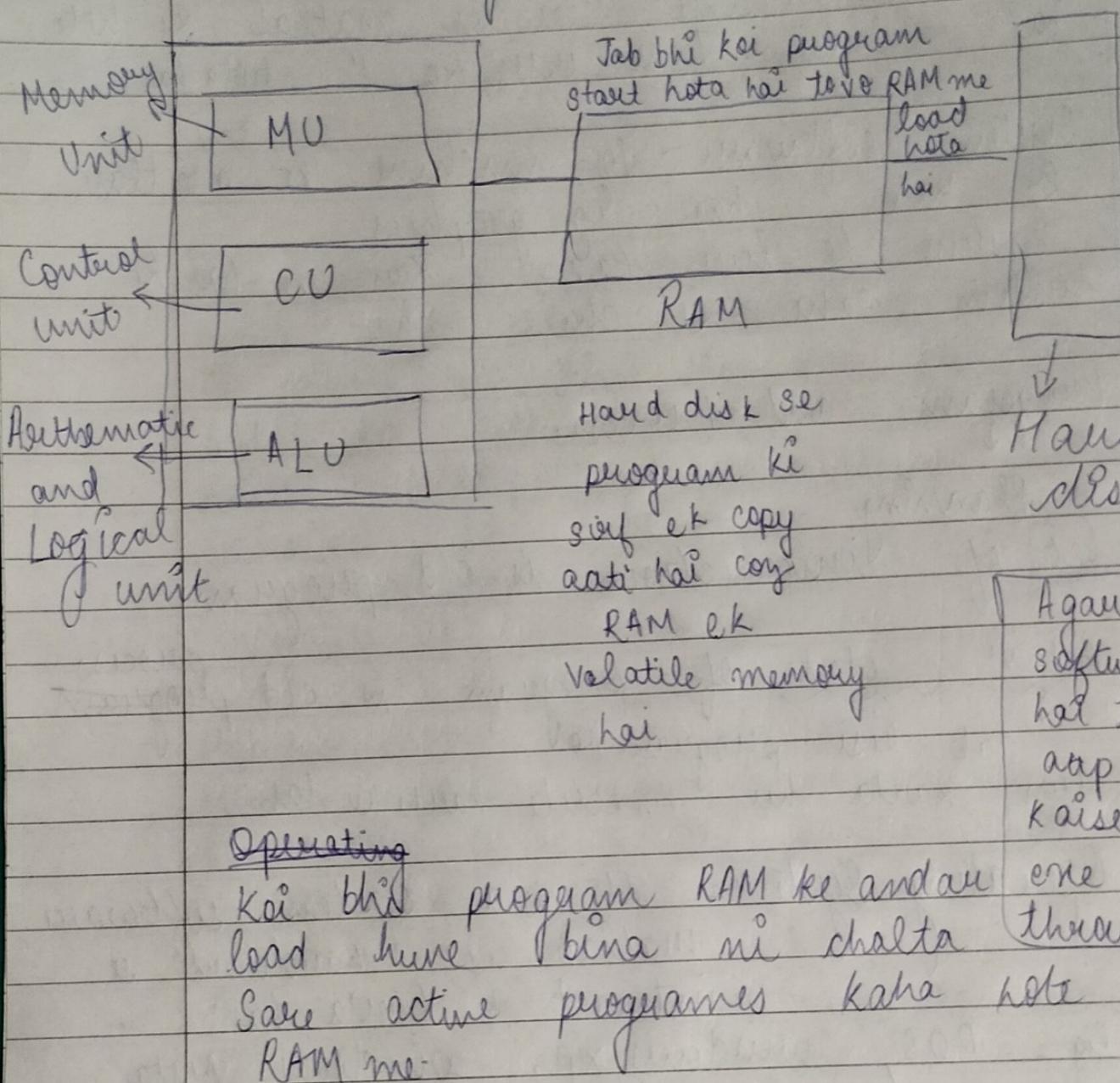
Eg = DOS, windows xp, windows vista,

2 Interface b/w user and machine

3 Act as manager of computer system

4 It does process management, memory management  
and file management

# Understanding execution of a program



Operating system ki jamedari hai ki vo program ke chalne ke liye memory mil jaee to "start ho jaiga"

Ek baar hamare program ko memory mil jaee to "start ho jaiga" mein ka execution start ho jaiga

Hamare program me kya hai? Bahut sare instructions

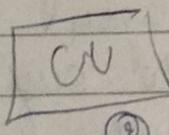
16 bit ke architecture ki hisab se first 0,1 hai jo hamara first instruction hai. Agle 16, 0 and 1 kya hai doosre instructions esse kai same instructions hamare program me likhi hui hain.

Ek instruction ka size fir hai 16 bites. Ab ye execute kaise hogi? Ye jaiga processor ke pass.

- ①  $\hookrightarrow$  Processor ke andar neta hai memory unit jaha par kai sare chote chote memory device hote hain.

$\boxed{MU} \rightarrow$  Set of registers

Tumse se ek register hota hai instruction register. Instruction register me instruction jahan store ho jaiga. Processor ke andar jo register store hota hai, vo bhi 16 byte ka hota hai. 16 byte matlab ~~isola~~ 0,1 issi me likha hai kya karna hai comp ko



②

$\boxed{CU} \rightarrow$  Circuit that reads instructions and decode it.

Ye samajta hai instructions ko ki kya karna hai?

- ③ CU, ALU ko signal dege ki ALU tumko ye kaam karna hai. ALU is vo circuit hai jo har taah ki calculation perform karta hai.

Ab ALU apni manji se kuch ni karega control unit (CU) usse jaisa bolega vo vaja ni ~~karega~~ karega.

$\boxed{ALU} \rightarrow$  Responsible for all arithmetical and logical calculations.

ALU uss instruction ko execute kar dega aur esse pheli instruction execute ho jaigi. Baum baum kanki sare instruction execute hongi issi tarah se. Tab sare instruction execute ho jaingi tab hamare program end ho jaiga.

### • SOFTWARE DEVELOPMENT IN C.

Software banate kaise hai uske bare me padenge. Hame banani hai ek ese file kyuki ese file hi ek software hai.

Eg apan ko banani hai ek file Sum.ece  
 Apni file ko banane me ye dikkat hai ki ye operating system dependent language hai matlab agar ye <sup>file</sup> doc operating system ke samajne layak hai to kisi another operating system ko samaj me nahi aaiga. To agar mai iss file ko banata bhi hu to suff ye doc operating system par chalegi aur kisi operating system par nahi". Jo ye operating system dependent code hai isliye c language me jab bhi software banaya jata hai to software particular operating system ke hisab se chalta ha. Jo agar mai iss file ke 0, 1 bhi seekhu tab bhi ye kisi ek operating system ke hi samaj me aaiga. Jo mujhe alag alag operating system ke liye alag alag coding seekhe padge. Isse aasan tarika hai ki hum kya seekhe c language.

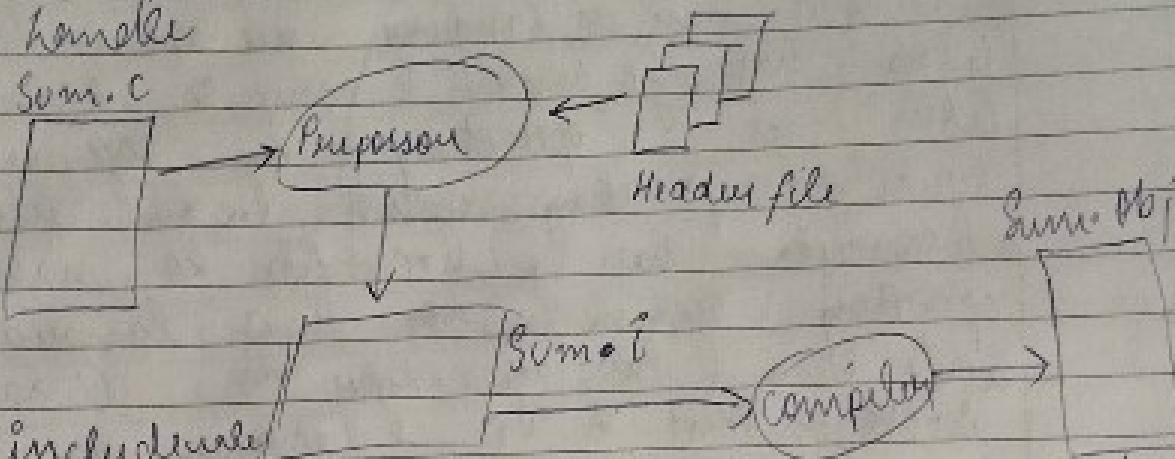
To apan kya karenge c language me coding  
 karenge to iss file ko kya bolengi source  
 file Iss file ka extension kya hogi .c  
 Ye file me c language me apan ne kya  
 likha hai, jo name comp se karna hai.  
 Lekin ye operating system software ni hai  
 kyuki operating system .exe file ko  
 samajta hai, .c file ko nahi. Yani  
 operating system ko c language ni  
 aati. C language kya hai hardware  
 independent. C language me program  
 banana aasan hai, user friendly way me  
 Ab jaanat hai ki hum iss c file  
 ko software me convert kare. Ab isko  
 convert karne ke liye name chalana hota  
 hai ek preprocessor software.

Preprocessor software hamne program me  
 jitne bhi statement hash/ se start ho raho  
 hain unko ~~start~~<sup>handle</sup> karne ka kaam iss software  
 ka nota hai

Hamne program me jitne bhi header files  
 ka zikr hota hai, unhy header files ko  
 hamne ni banaya hai ye files ke  
 ander ka code, hamne program me  
 judna imp hota hai. Hamne apne program  
 me kuch esse lines likhi hai jiski  
 vajah se preprocessor header files ke content  
 ko hamne file ke content me min  
 kar deta hai aur ek nai file banata hai  
 Ye mayi file hamne file jaisi hi hai  
 bas hash se start hone wale statement

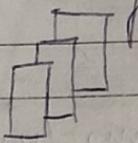
vaha se hoth chuke hai aur doosre statement  
 vaha aa chuke hai. Basically Preprocessor  
 se start heni wali statements ke aache  
 se hamle

Sum. c

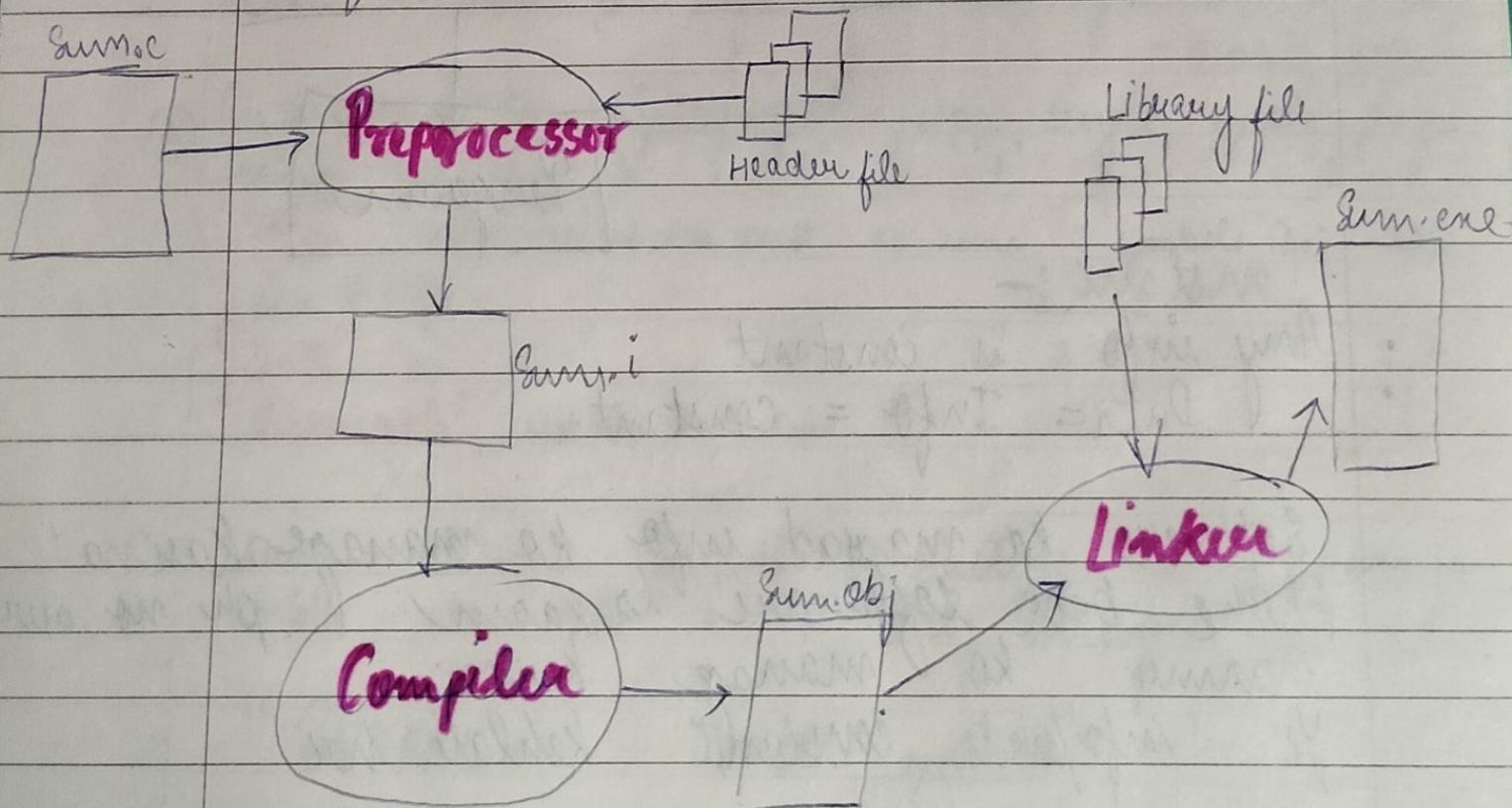


e.g. #include std.i.h Ye ek command hais,  
 jiska matlab hai ki Header file std.i.h  
 ki andar jo koi content hai usko progra  
 m me jod de to preprocessor std.i.h wali line  
 ke hatakou aur header file std.i.h me jo koi  
 likha hai wo apki program me shamil hau  
 dega. Is tarah wo code include ho jaya  
 jo header file me li kha hai. To file bani ab  
 usko hum compiler software ki madad  
 se hone kaunsa hai translate. Compiler hamara  
 program ko translate karke ek mai file bana  
 dega jo particular operating system ke  
 hisab se bani hoi. Compiler along along  
 operating system ke hisab se alay-alay  
 aata hai. Koi compiler dog ke hisab  
 se convert program ko to lines ke hisab  
 Compiler ne jo file bani hai uska  
 extension hai .obj. Jo naam humne  
 apni file ka rakha tha, compiler thi  
 vaha maina rakhega.

Apne program me use karne sare cheeze hai jo operating system ni samaj jaiga unkha matlab phle se library file me stored hai. library file acts like a dictionary library file.



Library file ka code aur obj file ke code ko link karni ke liye kya use karvengi? Liken liken done ka lik kaalke ek naya file bana dega. Aur iss kaalak se hamara software bana jaiga



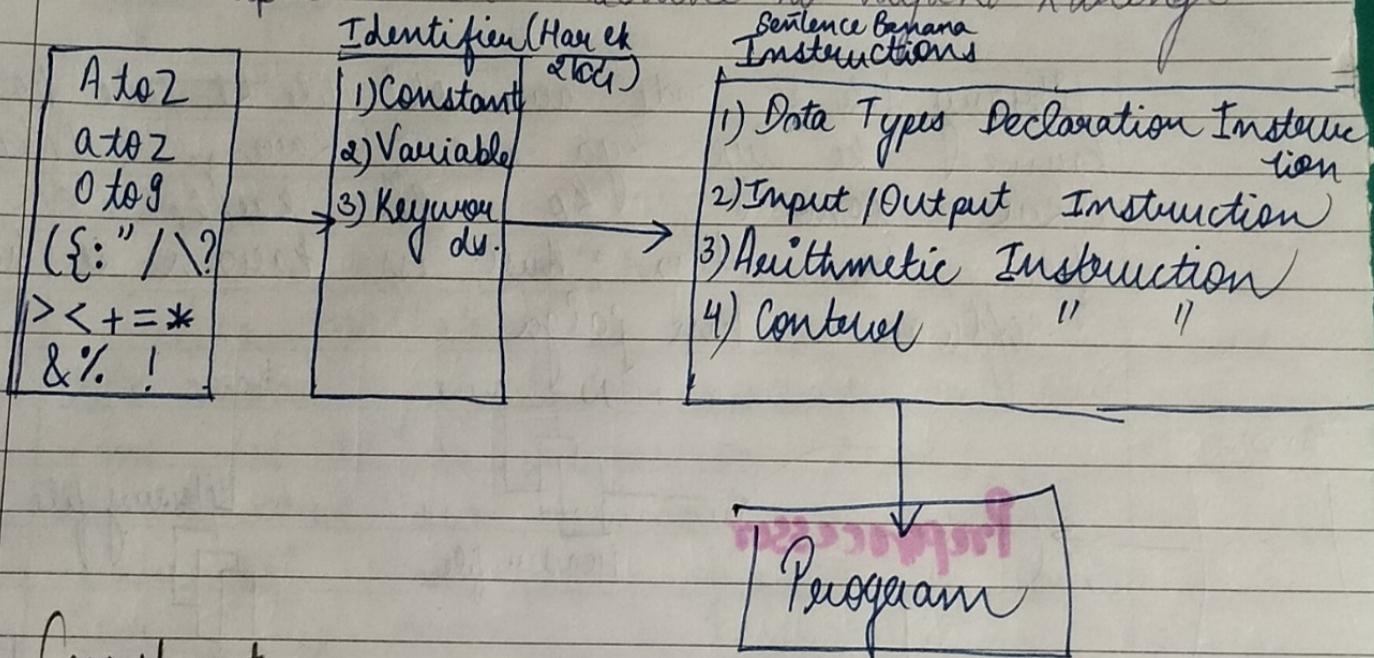
.c file koi software ni hai one file soft ka hai

par ap apne file ki file sum.exe karne ki kaise

Compiler software ki jaaniat nahi hogi. Humare asot ko na to compiler ki, na preprocessor ki na hi linker ki jaaniat hai. one file ka matlab hota hai executable file

### Identifiers

Hum apne vichar machine ko vayakt karenge



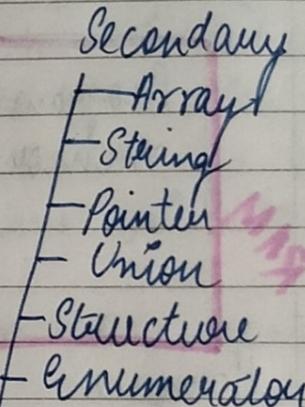
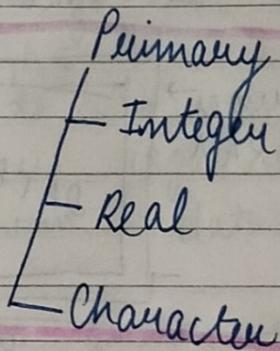
### Constant :-

- Any info is constant
- Data = Info = constant

Software ka magad info ko manage karna' phone book software ka karn ph no aur name ko manage karna. Ye info/data is constant kehlati hai.

## Types of Constants-

koi bhi software  
banate vakt  
nume pata hona  
mahiye hum kisi  
saah ki info ko  
manage kar raha hai



1. Integer constant - ek no. hai jo aur uss no. jisme point ni laga hua eg = -55, 25, 0

2. Real = Tinme pt laga hota hai. eg - 21.4, 3.56,  
- 0.065, 2.0

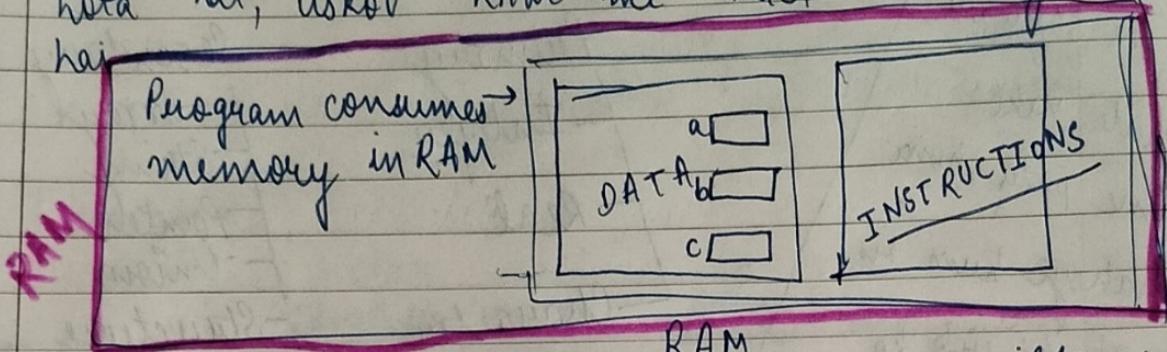
3. Character = koi bhi symbol jispe single, quote, laga hua ho. eg 'a', 'B', '+', '2', '2', '-3' Not a character const as isme 2 symbols hai - bhi aur 3 bhi.

'3.4' " " " " " 3 " " " 3, . 4  
**'Samabb'** Not a character

4. String = ek se jyada characters ko milakar jo info bani hai vo hai string eg  
"BHOPAL" Double quotes.

- Secondary constants vo constants hote hai jo primary ki Sahayata se bane hai.

Process Memory - Jab bhi hamara program run hota hai, usko RAM me kuch memory milti hai.



Hamare program ko <sup>RAM</sup> memory milti hai, ek instructions ko rakhne ke liye aur doosra data rakhna jaiga.

Kisi bhi program ke do ahen hisse hote hai, data and instructions.

ek command or nirdesh  
Program me instructions data to use karte hai.

- Ham apne program me ek khaas line likhkar compiler to batana hota hai ki hume kitni jagah ki jaaniyat hogi, data ko rakhne ke liye.

Iti - Aapko do nosr ko add karna hai, to aapko data do no ko handle karna hai aur uske result ko handle karna hai.

Aap jagah reserve karate hai. Ye reservation ke liye aap program me ek khaas line likhte ho jisse compiler ko pata chalta hai ki aapko kitni jagah ki jaaniyat hai aur aap utni jagah mangte ho jisme aapko data ko store karna hai. jab tak aapka program run ho raha hai. Ye jo

Memory me jagah hai, iske pachan ke liye naam diya jata hai jo apan apni marge se de sakte hai eg a,b,c to a,b,c hi variables kahlate hai, to variables ~~to~~ ka matlab Memory me uss jagah ke naam diya hua hai, jaha hum data rakhne wale hai

- Variables are the names of memory locations where we store data:

### Rules

- Variables name is any combination of alphabet, digit and underscore
- ~~Valid~~ A valid variable name cannot start with digit

A-Z or a-z

~~Case sensitive~~  
~~A and a diff~~

### Keywords:-

- What is identifier?

Hns Sabse chota meaningful shabd. Ek ek ~~ETC~~ ~~et~~ Sochiye, haan ek shabd ek identifier hai Compiler hamare program ke padhkar trans karega aur obj file banaiga. Compiler late ko hamare program ka haan ek ~~ETC~~ Samaj me aana chahiye tabhi vo usko translate kar paiga. Agar kuch ~~ETC~~ esse istamal kiye hai jaise variable ka naam, compiler errors digits ke matlab niv malum Isliye aap haan ~~ETC~~ ~~et~~ matlab khud batate hai. Lekin ~~et~~ kuch ~~ETC~~ esse

bhi hai, jinka matlab aapko compiler ko batane  
 ki jaannat ni hoti. ~~unke~~ <sup>kuch</sup> unka matlab  
~~unke~~ compiler ko phle se pata hota hai  
 esse ~~& TOC~~ \* Note hai keywords.

~~Key~~<sup>words</sup> = Pre defined words = Reserved words

C language has 32 keywords. (16 byte architecture)

auto	double	goto	signed	unsigned	variable
break	default	if	size of	void	name ke
case	enum	int	static	volatile	roop me
char	else	long	struct	while	nahi kare
continue	extern	register	switch		sakte
const	for	return	typedef		
do	float	short	union		

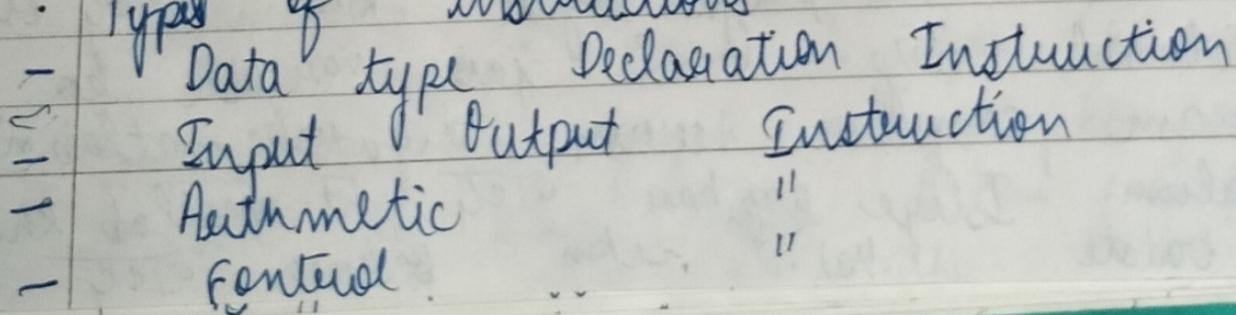
## DATA TYPE DECLARATION INSTRUCTION in C

Instruction -

Jaise hum statements bolte hai, vaise hum program  
 ing me unhe instructions bolte hai.  
 Hindi me hum unhe DIKHI Bolte hai

~~too~~

- Instructions are commands
- Program statements are C/D instructions
- Types of instructions



## DATA TYPE

- int
- char
- float
- double
- void

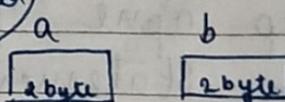
Esse ~~2~~ ~~10~~ Jo keyword thi hai aur datatype bhi, unhe num Primitive Data Type kehte hai.

### DECLARATION STATEMENTS -

Ab hum variables banana

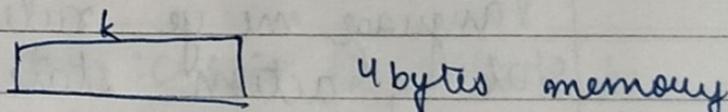
Eg assign  
int a, b = 5;

window based  
32 bytes  
4 bits



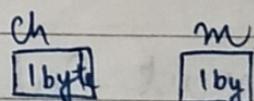
doc based  
2 bytes memory

float k;



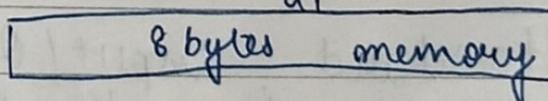
character  
constant  
store  
nota kai

char ch, m;



d1

double d,



Jab compiler apne program me esa dekhega to vo samaj jaiga ki ~~no~~ ye data type declaration instruction hai jiska phela ~~2~~ <sup>10</sup> ~~9~~ <sup>9</sup> Data Type hi hona chahiye. Ek line me aap ek se jayada data type ka istamal ni kar sakte. Compiler ke ye instruction milte hai ki kitne variable banane hai.

int = integer constant store  
float, double = real " "

Pt

a ko apne koi value assign ni ki to ro ineedi table value hai, garbage value Hum declaration ke baad bhi value initialise kar sakte hai Je jaenui ni nota jab aap variable

Ye declaration statements compiler ke liye hote hai variables ke naam ka pata chal sake. Puri program me jaha jaha ye statements aai to compiler ko meaning samaj aa jaiga.

Ye declaration statement hai, iske alawa jo bhi line aap apne program me likhte hai, unhe action statement kaha jata hai aur C language me ye rule hai ki haan block me declaration stat, action statement ke phle hi honi chahiye

## Input Output instruction in C language

Standard Input/Output device

- Keyboard is standard input device
- Monitor is " output "

- Output instruction ka matlab ek essa statement apne program me jiske chalne par screen par monitor par koi message print hota hai. Screen par kuch print karana hi output kehlaiga. Eise likhenge jisse output aa jal screen par. Print f is not a keyword print f is predefined function

function ek piece of code hai jisko apan ne  
namam de rakhा hai

PAGE NO. 10  
DATE: / /

Two types of messages

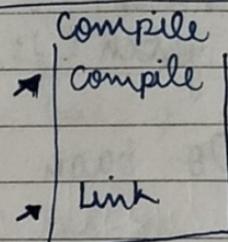
- Printing text as it is

- Printing value of expression or value of variable

main()

RA kaam  
ek choraas {  
keywoord }  
se harne  
printf("Saunub Shukla");  
getch();

Input instruction



Warning:

Link ka matlab hai one file ban gi

Click Run

getch ko use karte hai screen ko rokne ke  
liye

{

clrscr();  
printf ("Saunub Shukla");  
getch();  
}

main()

{

- clrscr(); // to clear screen → command \$ g++ compile n  
printf ("Saunub");  
printf (" Shukla").  
getch();

}

main()

```
clearscr(); // to clear the screen
printf ("Samabh\n");
printf ("Shukla");
getch();
```

In ki  
vajah se  
cheeska  
next line  
par aa jaiga

Do baar printf likhne ki jarurat ni hai

printf ("Samabh\n Shukla");

In kya kehata hai Escape Sequences

Special symbols jo \ se

eg \n = Next line                  start hotel hai

\t = Space aa jati hai beeth me

\b = MANPREEK AUR                  PEARKAUR

\\ = MANPREET\ KAUR

" = MANPREET " KAUR

\u = KAUR REET

Reet Kaur = KAURL  
Manpreetpearl = PEARLEET

Agar name centre me print karana hai to phir  
Output screen me 80 characters ek line me print ho saka  
lines 25. te  
hai

clearscr();

gotoxy(40,13);

printf ("Samabh");

getch();

main()  
{

Declarer  
Action

```
int a=4, b=5;
close();
print f ("a");
getch();
```

("4") Ye galat hai vo  
4 hi print kar  
raka hai, a ki value  
ni

Agar a ki value print karanai hai to

printf ("%d", a);

Ab a ki value aaigi

%.d ek special symbol hai.

/\*

Format Specifier bolte hai

%.d int

%.f float

%.c char

%.lf double

printf ("a=%d", a);

"Value of a is %.d", a);

printf ("Value of a is %.d  
and b is %.d", a, b);

printf ("Sum of %.d and %.d is %.d", a, b, a+b);

Scarf()

Scarf() is not a keyword

Scarf() is a predefined function

PAGE NO. icon

DATE: / /

&  
Add

main()

{

int x;  
clrscr();  
scanf ("%d", &x);  
getch();

Keyboard se vo  
value lega

Scarf bhi eco  
karta hai

Keyboard se  
kuch data lana  
hai to Scarf ki  
jammat padegi

Screen  
clear  
auto run

scanf ("%d", &x);

printf ("Square of %d is %d", x, x\*x);

aur phir usko  
batai gai variable me store  
karna.

main()

{

int n;

clrscr();

printf ("Enter a number");

scanf ("%d", &n);

getch();

Jeet ke  
Karna  
kyo hais

main()

{

int x, y;

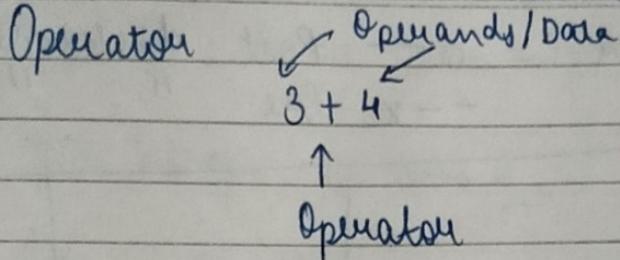
clrscr();

printf ("Enter two numbers");

scanf ("%d %d", &x, &y);

printf ("Sum of %d and %d is %d", x, y, x+y);

# Arithmetic Instruction in C language.



## Arithmetic Instruction

- An instruction which is used to manipulate data using operators, is known as Arithmetic Instruction
- There is no BODMAS in C language.

e.g.)  $3 + 4 * 5$

## Operator types

- Unary Operators
- Arithmetic "
- Bitwise "
- Relational "
- Logical "
- Conditional "
- Assignment "

main()

```

{
    int x=3;
    clrscr();
    x++; // x = x + 1
    printf("%d", x);
    getch();
}
  
```

Run = 4

Pre increment

$\downarrow$   
++x likho  
ya x++ same  
baat

Post  
increment

$\downarrow$   
Sign batane wale  
 $\downarrow$   
++  $\Rightarrow$  increment operator

{  
int x=3;

x ++;

printf ("%d", x);

$\downarrow$   
++x  
printf ("%d", x); , getch();

Run = 45

$\uparrow$   
Dosee baar  
ek baar } add kya  
phle }

Post increment ki priority sabse kam hoti hai

Pre increment " " "

*panchayat icon*  
DATE: / /

- Decrement operator       $x--$  Post decrement  
                                 $--x$  Pre " "

{

int  $x = 3, y;$

$y = x++;$

printf ("%d %d", x, y);

getch();

}

~~#~~ Run = 34

$y = ++x$

$y = --x$

$y = x -$

Run  
Run = 44 22

Size of ()

*eg*

main()

{

- Size of (data type)
- Size of (variable)
- Size of (constant)

L5, P3

Bytes

int  $x;$   
 $x = \text{size of (float)};$   
printf ("%d", x)  
getch();

}

Run<sup>u</sup>

main()

{

int  $x, y, z;$   
 $x = \text{size of (34)};$   
 $y = \text{size of (3.56)};$   
 $z = \text{size of ('a')};$   
printf ("%d.%d.%d",  
      x, y, z);  
getch();

}

282

(double) 8  
(int) 4  
(char) 1

Run<sup>u</sup>

*eg* {

int  $x, y;$   
float k;  
double d1;  
char ch;  
clrscr();

$x = \text{size of (y)};$   
printf ("%d", x);  
getch();

y ek variable  
hai, jiska type  
int hai

Double ka size jayada  
nota aachhi se decimal ko

(K) 4

$z = \text{Size of } ('a');$  // ASCII

04255  
Date

character constant

Han ek character constant ka ek code hote hai  
a ki ASCII value = 97

PAGE NO. 12

DATE

97 integer to ans 2 Comp 4

### - Arithmetic Operations

\* / %  
+ -

Priority bracket hain tere li  
left to right solve kipa jiga

3 + 4 - 7 To phle + bas me -

Priority same

main()

{

int x;

x = 3 + 4;

printf ("%d", x);

getch();

gy

$x = 3 - 4$  Run = -1

$x = 3 * 4$  Run = 12

$x = 3 \% 4$  Run = 0 not 0.75

language ka rule hai do integers ke beech me operator perform karne ke ans lise

Modular Oppe

Integer hi aiga

$x = 37.4$  remainder  $x = 1/5$

Run 3 data hai

Run = 2

Run = 1

Real Result

$x = 137.4$  Run 1

$x = 237.5$  Run 3

$x = 251.5$  Run 0

ek bhi real hi to result real

aiga

some integer hai tabhi result integer aiga

$3/4$  0

$3.0/4$  0.75

$3/4.0$  0.75

$3.0/4.0$  0.75

visibility kise check hogi?  
operator after se

# BITWISE OPERATORS

Bitwise AND &  
Bitwise OR |  
Bitwise XOR ^  
Bitwise NOT ~  
Right Shift >>  
Left Shift <<

Exclusive OR

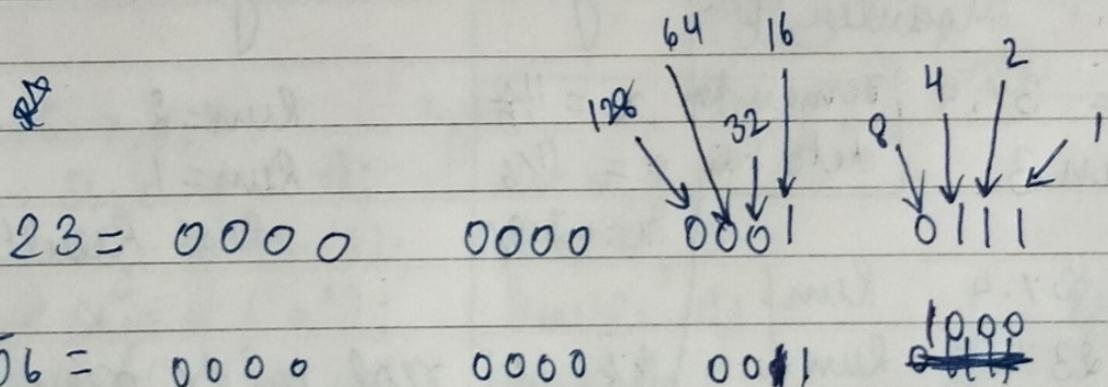
Sirf  
Daur / pe  
kaam kante  
hai

& operator	Operator	^ Operator
$0 \& 0 = 0$	$0   0 = 0$	$0 ^ 0 = 0$
$0 \& 1 = 0$	$0   1 = 1$	$0 ^ 1 = 1$
$1 \& 0 = 0$	$1   0 = 1$	$1 ^ 0 = 1$
$1 \& 1 = 1$	$1   1 = 1$	$1 ^ 1 = 0$

int x;  
x = 23 & 56;

Rum = 0000 0000 0000 10000 = 16  
23 = 0000 0000 0001 0111  
56 = " " 0011 1000

Yaha par sola bits hai to ek ek ke correspond  
sola ban and operation perform karenge.



Eg)  $x = 23 | 56$   
Rum = 0000 0000 0000 0011 1111 = 63

Ex)

```
main()
{
    add();
    getch();
}
```

/\* Takes Nothing Returns Nothing \*/

PAGE NO. 20  
DATE: / /

```
define
kiya hai
{
    int a, b, c;
    printf("Enter two numbers");
    scanf("%d %d", &a, &b);
    c = a + b;
    printf("Sum is %.d", c);
}
```

Agar koi function takes nothing wale nature ka koga to uske parenthesis empty honge call karte huye bhi aur define karte huye bhi

scanf

Lekin printf, ko call karte samay parent empty ni hain to ye takes some <sup>thing</sup> ka hain.

getch() ko call karte hain tab parenthesis khali hain to ye takes nothing nature ka hain.

Return kaunsa ka matlab add function ko fisre call kiya hai, usko value layjana

Declaration statement compiler ke liye hote hain jisse ki compiler ko un shabdo ka matlab pata chal sake.

koi function agar undefined hai to uska <sup>declaration</sup> kisi na kisi header file me hoga

include <conio.h>

↓  
console input output

include <stdio.h>

↓

standard input output

Header files

Han function ka declaration kuch ~~compiles~~ Khud  
 kar dete hai like prefer hame kaun chahiye  
 declaration kaun ka  
 Takes nothing Returns Nothing  
 void main()

{  
 function void add (void); empty  
 declaration

close();  
 add();  
 getch();

return;  
 return type void add ()  
 {

int a,b,c;  
 printf ("Enter two numbers");  
 scanf ("%d %d", &a, &b);  
 c = a + b;  
 printf ("Sum is %d", c);

Han function  
 ke apne  
 variables  
 hote hain

Doosre functi  
 ki values <sup>(ph)</sup>  
 ko le sakte  
 hain

Variables excess  
 ni kar sakte

- Takes something, returns nothing -

void main ()

{  
 int x,y;  
 void add (int, int);  
 prn

Takes ka matalab  
 parenthesis  
 me likhi  
 baat se  
 sign  
 agar aur  
 returns ka  
 void se



## TAKES NOTHING, RETURNS SOMETHING

```
int add(void);
void main();
{
```

```
    int s;
    clrscr();
```

```
s = add();
```

```
printf ("Sum is %d", s);
```

```
getch();
```

```
} int add()
```

```
{ int a, b, c;
```

```
printf ("Enter two numbers");
```

```
scanf ("%d %d", &a, &b);
```

```
c = a + b;
```

```
return (c);
```

```
}
```

↓  
Return apan c ko nahi  
balki uski value ko  
kai raha hai

c kaisa hai? int  
nature ka to apan  
ne int add  
likha hai.

Ye c ki value kaha return hogi? Jaha par  
apan ne add function ko call kiya hai  
ie main ne call kiya tha to s  
ke paas c ki value aa jaigi

Return ki madad se apan ek hi value return  
kara sakte hai. Ek se jayada ni. Kyu kyuki  
jo value return karoge vo function call ki jag

Let us assume apna me  $c = a+b$  ni likha aur  
seeta return  $a+b$  likh diya to  
value return ho jaiga agar PAGE NO. & DATE: / /

Sum ki value return karani hal to phir apna  
return ( $a+b$ ) likhna padega.

Return ki madad se na kewal value return hoti hai,  
balki control bhi return ho jata hai.

Assume return ( $a+b$ );  
printf ("HELLO");

Hello print ho hi ni paiga kyuki & return ke  
baad kaam hi khatam ho jaiga. Return  
chal gaya to uske baad ki koi aur line  
ni chalgae.

TAKES SOMETHING, RETURNS SOMETHING

int add (int, int);

void main()

{

    int x, y;

    clrscr();

    printf ("Enter two numbers");

    scanf ("%d %d", &x, &y);

    s = add (x, y);

        value pass ho jaiga

    printf ("Sum is %d", s);

    getch ();

}

    int c;

    add (int a, int b),

    c = a + b

    return (c);

}

# RECURSION

PAGE NO. / /  
DATE: / /

- Function calling itself is called Recursion  
Apan jante hai koi bhi function, kisi bhi function ko call kar sakta hai.

Eg) main()

{  
int k;

Actions  
Statement  
} k = fun(3);  
printf ("%d", k);

int fun (int a)  
{

int s;  
if (a == 1)  
return (a);  
s = a + fun(a-1);  
return (s);  
}

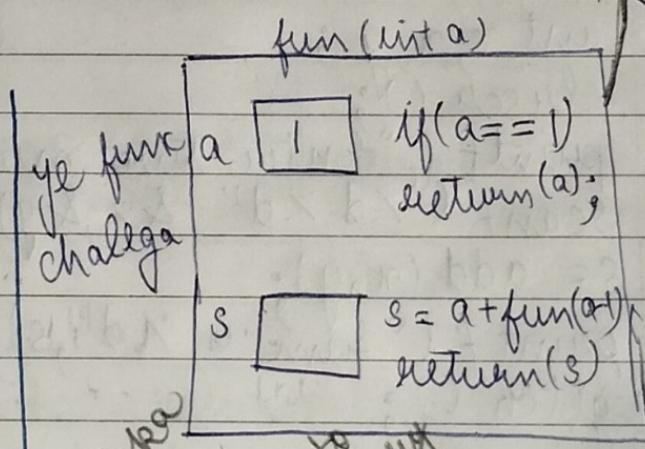
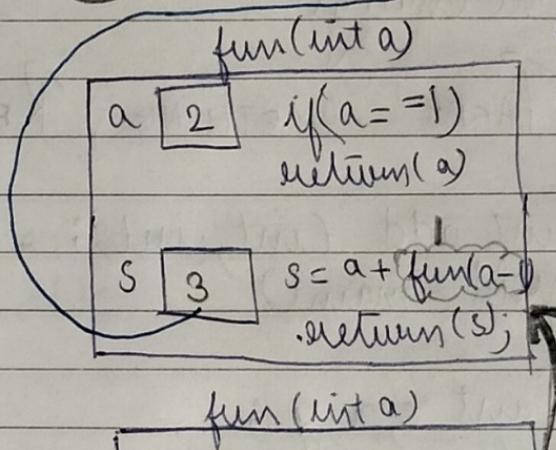
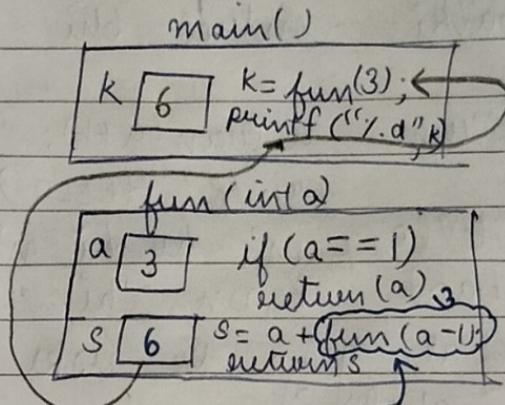
Sum of n natural nos.

$$\text{Sum}(10) = 10 + \text{Sum}(9)$$

$$\text{Sum}(9) = 9 + \text{Sum}(8)$$

Badi problem ko solve karne  
hai to recursion ki help  
Se aasani se solve ho  
jaiga

Prob ki complexity  
to reduce karne  
rake hai.



Tekra  
karne  
khatam to  
useki memory  
release ho  
jai gi.

No function khud ko baar baa�  
call kar raha hai to ek pt par khatam  
bhi karne hai to dead end, jawaab hai

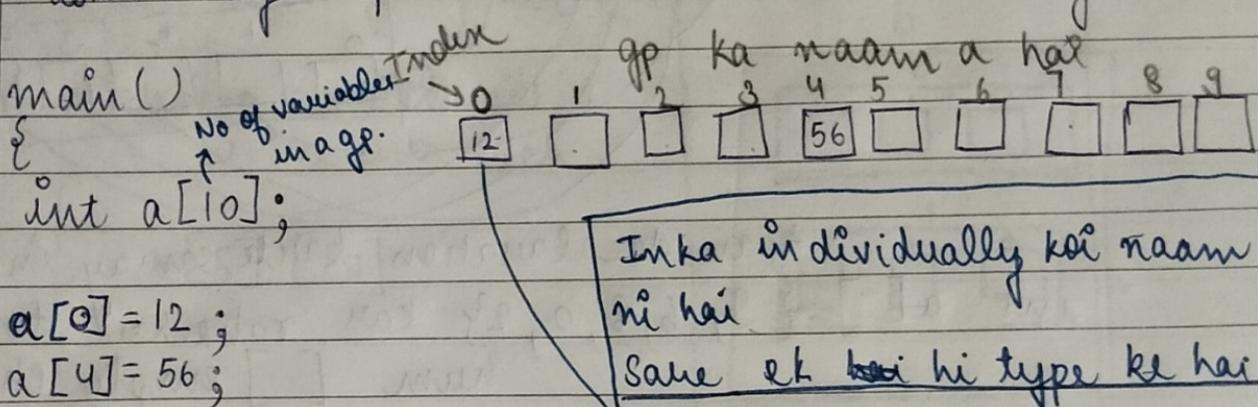
[ ]  
Sift away ke  
use me use  
range.

## ARRAYS

- Array is a linear collection of similar elements.
- Array is also known as Subscript Variable
- " " a gp of variables

Eg hume 100 Variables ko add karna hai to likte time  
100 variables declare karna padega phir 100 bar  
1. d likhna padega to bahut dikkat ho jaigi  
iske liye apan use karte hai, array

eg)



eg

main()

Avg may  
be in  
point

```
int a[10], i; sum = 0;
float avg;
printf ("Enter 10 nos");
for (i=0; i<=9; i++)
scanf ("%d", &a[i]);
```

```
for (i=0; i<=9; i++)
sum = sum + a[i];
```

```
avg = sum/10.0;
```

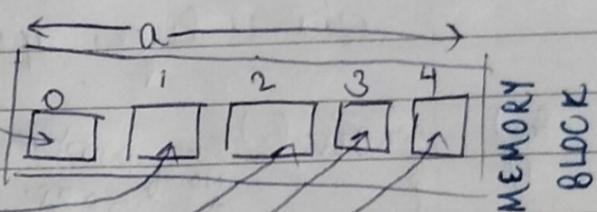
```
printf ("Average is %.2f", avg);
```

```
getch();
```

{}

## Declaration in ARRAY

- `int a[];` Agar empty hai to compiler aa jaiga coz compiler ko pata ni chalega, kitne variable hai.
- `int a[5] = { 9, 6, 8, 0, 2 };` Assume  $a[0]=9$  to  $a[1]$  me assign  
 ↳ Index nahi hai ye total no. of variables hai
- `int a[5] = { 9, 6, 8, 0, 2 };`



Apan declaration karte samay initialise

- `int a[] = { 9, 6, 8, 0, 2 }` kar rakte hai to phir hum [ ] empty choose sakte hai.
- Array lagana memory consume kerte hai

e.g.) `int a[5] = { 9, 6, 8, 0, 2, 4, 7 };` error

`int a[5] = { 9, 6 };` Bache huye blocks me o aa jaiga

→ ye condition tabhi kaam kangi fab declare karlete huye ni initialisation kara ho.

# TWO DIMENSIONAL ARRAY

- $\text{int } a[6] \rightarrow \text{One D Array}$
- $\text{int } b[2][3] \rightarrow \text{2 D Array}$   
 $2 \times 3 = 6$

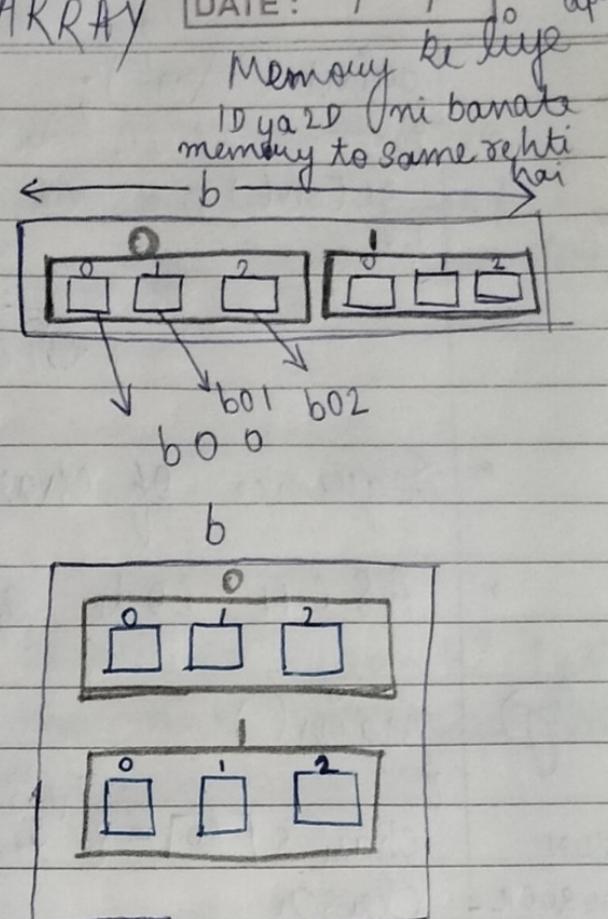
Array of arrays.

Q Why 2D Array?

Bahut sare grouped data  
ke liye

Eg 1 School st. 5 classes  
10 student each

$a[5][10]$ .



Eg) main()

```

int A[3][3], B[3][3], C[3][3], i, j;
clrscr();
printf("Enter 9 numbers for first matrix");
for(i=0; i<=2; i++)
    for(j=0; j<=2; j++)
        scanf("%d", &A[i][j]);
printf("Enter 9 nos for second matrix");
for(i=0; i<=2; i++)
    for(j=0; j<=2; j++)
        scanf("%d", &B[i][j]);
for(i=0; i<=2; i++)
    for(j=0; j<=2; j++)
        {
            C[i][j] = A[i][j] + B[i][j];
            printf("%d", C[i][j]);
        }
    
```

```

    cout ("\\n");
}
getch();
}

```

## STRINGS IN C

- Sequence of characters terminated at null character [lctn]
- ASCII code of null character is 0 (zero)

Ex)

main()

{  
char s[10] = { 'S', 'A', 'U', 'R', 'B', 'H', '\0' };

RUN  
SAURUBH -

class

for (i=0; i <= 7; i++)

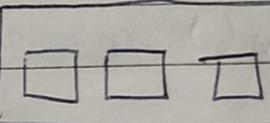
printf ("%c", s[i]);

getch();

} character ko print karna

OR

hai



Null ki

madad se,

string me

kitne character

hain hain Jane

ki jaune ka

ni. padti

main

{

char s[10] = { 'S', 'A', 'U', 'R', 'B', 'H', '\0' };

int i;

Array ke block

for (i=0; s[i] != '\0'; i++)

printf ("%c", s[i]);

getch();

}

Print string ko ek baar me print

print ("%s", s);

getch();

ek character  
ko jab  
examine  
karna hoga

Print  
string  
ko print

printf ki jagah puts bhi use kar sakte hai.  
 puts sirf string ko print karne ke kaam nahi karta hai

PAGE NO. 23

DATE: / /

puts(s);

RUN = SAURABH

getch();

cursor new line pe aa jaiga

puts apne aap line change karne deta hai lekin printf me phle line change karne ke liye special characters use karne padte hain. eg /n.

char me string ke tarika

eg) main()  
{

char s[10] = "SAURABH"; Compiler apne aap ke last me null character jood dega.

User se bolte hai naam enter karne ke liye.

main()  
{

char s[20];

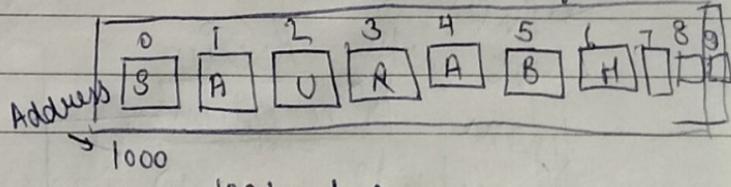
int i;

printf ("Enter your name");

scanf ("%s", s); // s=1000 =&s[0]

puts (\*s);

getch();



Array ka naam  
bhar likhna  
use first block  
ke address ke  
baare me  
bataega

Address  
of likha  
hai

getch();

printf

Scanf ek se jayada string ka input ni  
le sakta

s likhe ya  
&s[0] ek

scanf ke laga  
SAURABH SHUKLA data khatam baat  
Run ignore ho gaya

delimited by  
No space  
No enter

AMIT  
RUN  
AMIT

Apar scarf ki jagah gets(s) use karne hai vo icon  
 do string ko le lega  
 gets input lene ka kaam karta hai !

PAGE NO.

DATE:

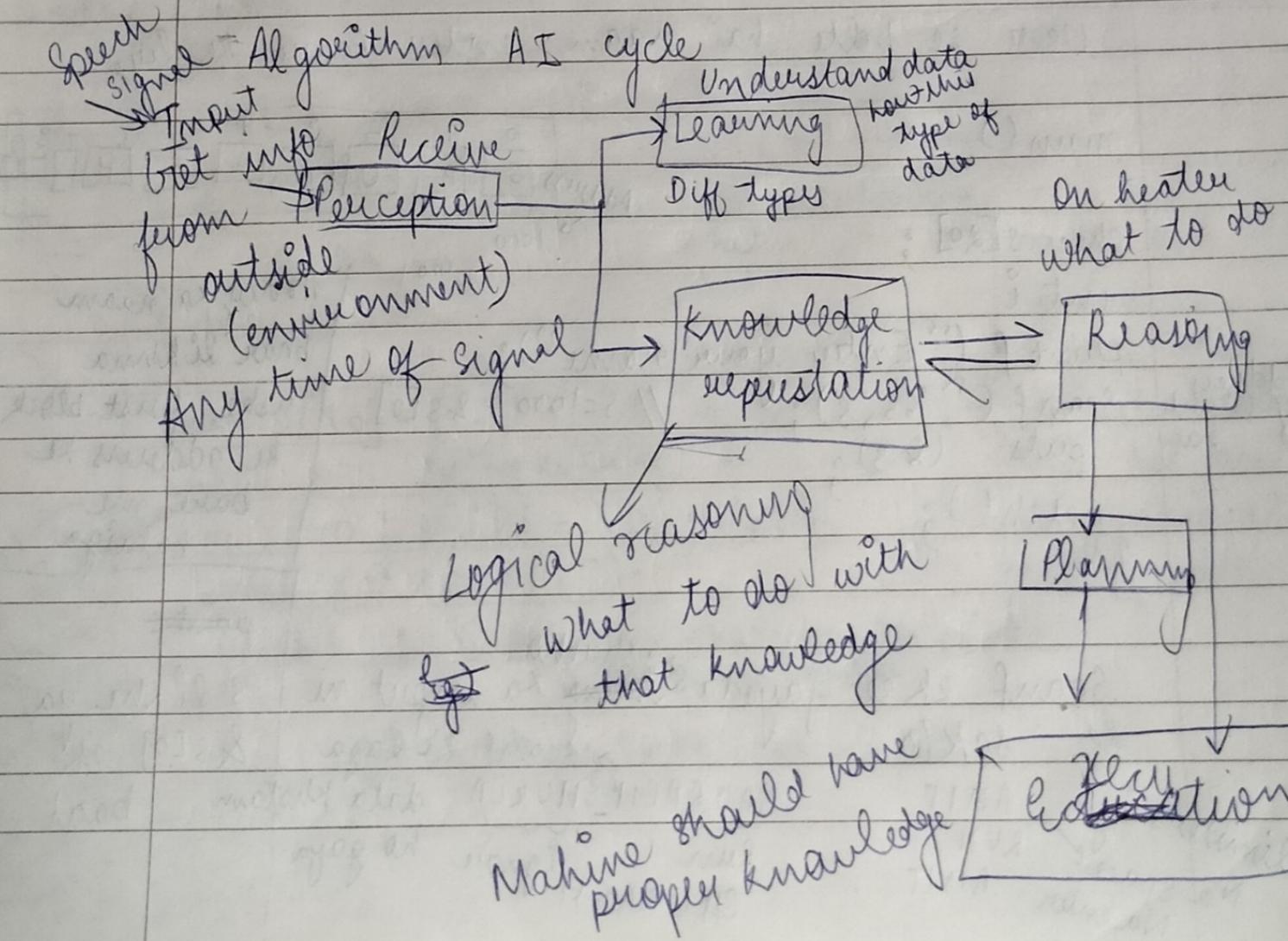
ye space ko delimiter ni manta  
 in place of scanf;  
 gets (s);

Intelligence  
 ↓  
 Prob solving  
 reasoning

Artificial  
 2 layers

Neurons  
 Neural network  
 diff type

# Machine should be intelligent



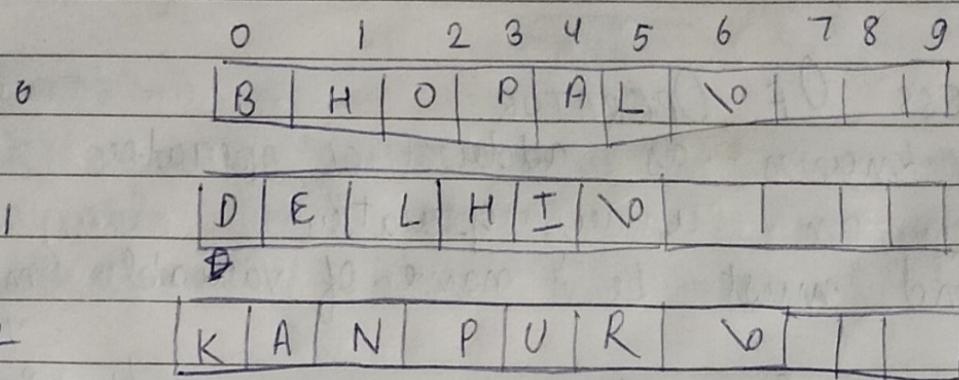
## String related function

- strlen()
- strrev()
- strlwr()
- strupr()
- strcpy()
- strcmp()
- strcat()

Programming inc

### - Handling Multiple Strings

- char s[3][10] = { "BHOPAL", "DELHI", "KANPUR" };



Eg)

main()

{

char s[3][10];

int i;

printf("Enter three strings");

gets(&s[0][0]); for (i=0; i<=2; i++)

gets(&s[i][0]); // gets(s[i])

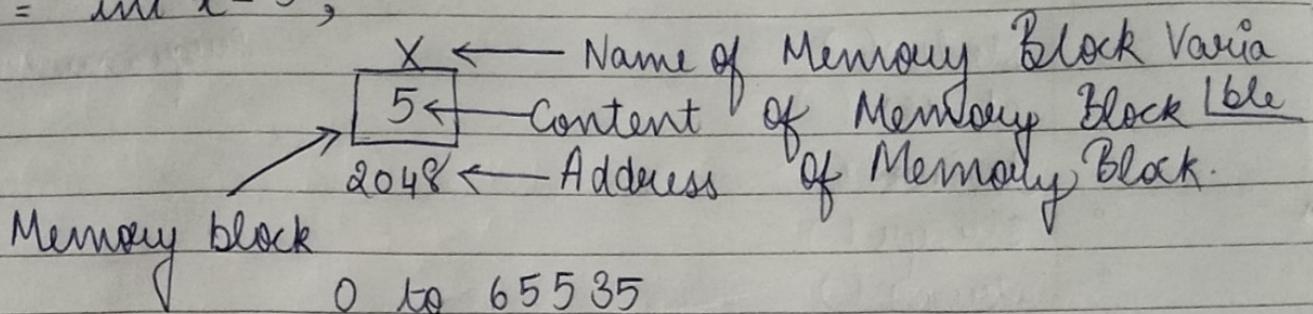
for (i=0; i<=2; i++)

printf("%s\n", s[i]);

getch();

## POINTERS

Eg = int  $x = 5;$



main()

Run 5

{

2048

int  $x = 5;$

printf ("%d\n",  $x$ );

printf ("%d", & $x$ );

}

## ADDRESS OF OPERATOR

- & is known as address of operator
- It is an unary operator
- Operand must be name of variable (matlab &  $x$ )  
variable name
- & operator gives address of number  
of variable  
chahiye not constant
- & is known as referencing operator

## INDIRECTION OPERATOR

- \* is indirection operator  
dereferencing
  - Unary operator
  - \* It takes address as an argument
  - \* returns the content / container whose address is its argument
- Address like  
Variable ka naam  
bata data hai

Eg) main()

```
int x=5;
printf ("%d\n", x);
printf ("%d", &x);
printf ("%d", * &x);
```

Run

5
2048
5

%d - 32768 to 32767

%u 0 to 65535

Ques)

```
int x=5;
&x=7;
```

x
5
2048

Error

We cannot store anything in &x as &x is not a variable, it is the way to represent address of block x.

Address ke saath me constant ni likh sakte, left me to likh sakte hai.

Eg) j=&x

x	j
5	2048
2048	3000

We can store address in another variable  
But j has to be declared before use.

→ Ye such symbol hai

int \*j Apan ne compiler ko bata diya hai ki  
j special tarek ka hai jisme address hai

- j is not an ordinary variable like any other integer variable
  - It is a variable which contains address of another variable.
- PAGE NO.   icon  
DATE: / / address
- POINTER**

- Pointer is a variable that contains address of another variable.
- Pointer always consumes 2 bytes in memory.

Ques) main()

```
int x=5, *j;
```

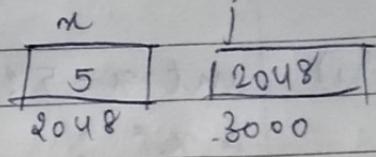
```
j=&x;
```

```
printf ("%d\n", x, j);
```

```
printf ("%d", *j, &x);
```

```
printf ("%d", *j);
```

```
y = cancel;
```



Asterisk aur imprecent saath me aai to cancel kar do.

Pointer se phle \* aai to j ye punar vo ban jaiga jiski taraf apan point karna chah raha hai.

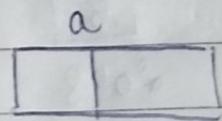
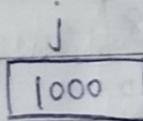
Run      5    2048

              5    2048

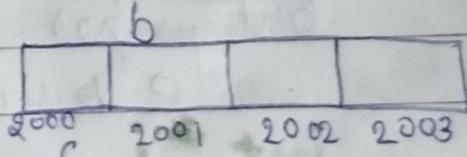
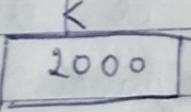
              2048.

### BASE ADDRESS

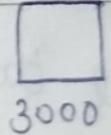
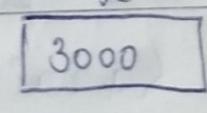
```
int a, *j;
```



```
float b, *k;
```



```
char c, *r;
```



★ Variable ke phle bigte ke address ko base address  
Kehte hai

PAGE NO.	26
DATE:	/ /

Pointer me sif base address hi jata hai.

e.g.)  $j = \&a$     $k = \&b$     $l = \&c$

$j$  ako point kau eagna hai.  
 $k$  bko "  
 $l$  c" "

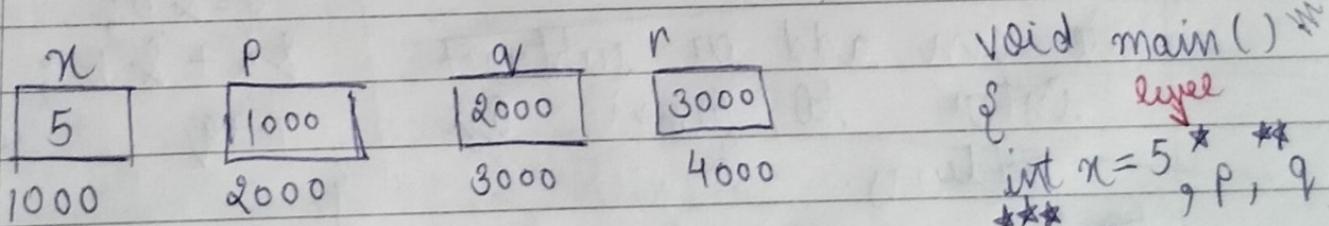
Jab hum \* $j$  likhenge to vo a hogा.

Pointer ka data type uske size par asav ni dalta  
lekin pointer me kis tarah ke variable ka  
address store hogा uss par asav dalta hai

**RULE** Pointer jis type ka hota hai uss type ke variable  
ko point karta hai

K ko wo sif 2000 pata hai vo float type  
ka hai isliye usko aage ke bits ke bare  
me bhi pata chal jaiga.

## ENTENDED CONCEPT OF POINTERS



$x$  ka  
maki  
 $m$  ka

lik hne kam ka hi save kar  
sakte hai jaise  $p$ ,  $x$  (lsv)

ka hi address store kar saka  
hai.

$$p = \&x;$$

$$q = \&p;$$

$$r = \&q;$$

$x$  is a int variable.  $p$  is the pointer to an int  
 $q$  is a pointer to a pointer to an int  
 $u$  is a pointer to a pointer to a pointer to an int

\*  $q = p$

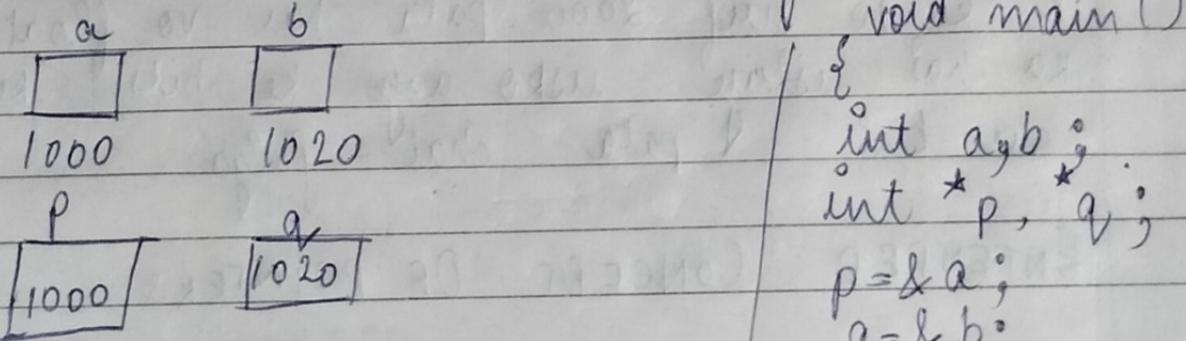
\*\*  $q = 7;$        $\boxed{x}$   
                         7  
                         1000

\*\*\*  $u = 7$

## POINTERS ARITHMETIC

• We cannot add, multiply or divide two addresses  
 (Sub is possible)

• We cannot multiply an integer to an address  
 and similarly we cannot divide an  
 address with an integer value



We can add or subtract  
 inter to / from an  
 address.

{  
 int a, b;  
 int \*p, \*q;  
 p=&a;  
 q=&b;  
 &a + &b; we cannot  
 p + q      Wrong add

~~$\cancel{p+1}$~~        $1002$   
 $p - 1 = 998.$

Pointer +  $n$  = pointer + size of (type of pointer)  
 $p + 4 = 1000 + 2^4$

- We can subtract two addresses but they should be of same type.

Possible  

$$q-p \quad 1020 - 1000 = 20 / 2 = 10$$
  

$$p-q \quad -10$$

- Pointer - pointer  $\neq$  Literal subtraction / size of type of pointer

### APPLICATION OF POINTERS:-

#### ACTUAL AND FORMAL ARGUMENTS.

function  
to declare  
kaise kai  
main()

```
void fun(int x, int y);
```

{

int a, b;

...

fun(a, b);



}

Actual argument

a	b
10	20

Formal  
arguments

x	y
10	20

void fun(int x, int y)

{

...

}

Lekin jab  
apni x  
aur y  
me change  
karenge  
to a, b me  
ni hoga.

32 bit int add 2\* number

64 bit

add 4\* number

by  
Am)

Write a function to swap two integers.

void swap (int\*, int\*);

{

PAGE NO. icon  
DATE: / /

function  
call by  
address

int a, b  
clrscr();  
printf ("enter two nos");  
scanf ("%d %d", &a, &b);  
printf ("a=%d b=%d", a, b);  
getch();

void swap ()

{  
int t;  
t = a;  
a = b;  
b = t;  
}

galat  
hai

value  
change  
kaise kuch  
ni hoga

a ki value aai hai is me  
void swap (int x, int y)

{  
int t;  
t = x;  
x = y;  
y = t;  
}

x aur y me change kare se a aur b  
me kuch change nہ hoga.

To apan ko seedha a aur b ko hے excess karna  
padega.

void swap (int \*x, int \*y)

{  
int t;  
t = \*x;  
\*x = \*y;  
\*y = t;  
}

# CALL By REFERENCE

PAGE NO. / /  
DATE: / /

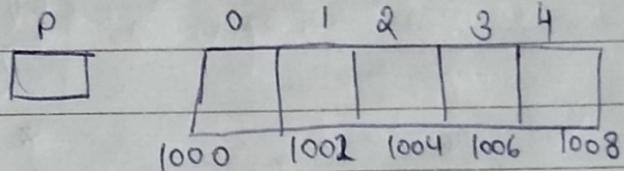
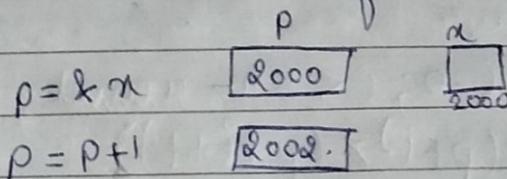
- Call by reference is same as call by address
- When formal arguments are point variables, it is call by reference
- Reference means address

Ques Why we use address of (&) in scanf?

Ans. - scanf is predefined function : main ek function hai, uske variable hai a, b to scanf a, b ko access ni kar paiga to isliye apna & likha hai.

## POINTERS AND ARRAY

- Pointers when incremented, always point to immediate next block of its own type



Ex)

main ()  
{

int a[5], \*p;

p = &a[0];

for (i=0; i<=4; i++)

scanf ("%d", &a[i]);

for (i=0; i<=4; i++) p++

printf ("%d", \*(p+i));

pointer se array ko point karte saare blocks ke access kartake saare blocks ke access kartake hain

```
main()
{
    void input (int *p)
    {
        int i;
        for (i=0; i<=4; i++)
            scanf ("%d", p+i);
    }

    void display (int *p)
    {
        int i;
        for (i=0; i<=4; i++)
            printf ("%d", *(p+i));
    }

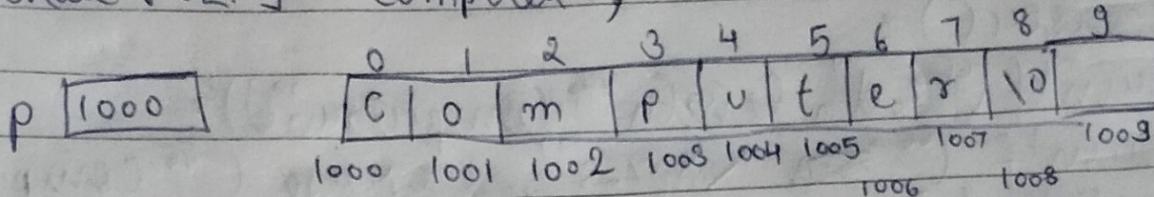
    void sort (int *p)
    {
        int round, t, i;
        for (round = 1; round <= 4; round++)
            for (i=0; i<=4-round; i++)
                if (*(p+i) > *(p+i+1))
                {
                    t = *(p+i); // a[i] > *(p+i)
                    *(p+i) = *(p+i+1);
                    *(p+i+1) = t;
                }
    }
}

main()
{
    int a[5];
    clrscr();
    input (a);
    display (a);
    getch();
}
```

## POINTERS AND STRING

~~X~~ Chan pointer

- String is stored in char array
  - char s[10] = "Computer";



- Chan pointers can point to chan block  
chan  $\star p$ ;  $p = \& s[0]$ ;

## ~~STRING~~ STRING CONSTANT

- String Literal = String Constant = String

printf ("Computer");  
char s[10] = "Computer";

• printf (s);

- printf (s)
- strlen (s)      String ki length calculate karne ka string hoti hai

~~stehlen (& s[0]);~~ <sup>die</sup> Address pass kann raus haben.

on  
steles ("Computer")

$$*(p+i) = S[i]$$

```
int length (char *);  
char * reverse (char *);  
main()  
{
```

```
    clrscr();  
    printf ("%d", length ("Computer"));  
    printf ("%s", reverse ("Computer"));  
    getch();  
}
```

char \* reverse (char \*p)

```
int l, i;  
char t;  
for (l=0; *(p+l) != '\0'; l++);  
for (i=0; i < l/2; i++);  
{  
    for (i=0;
```

$$\begin{aligned}t &= *(p+i); \\*(p+i) &= *(p+l-1-i); \\*(p+l-1-i) &= t;\end{aligned}$$

}

return (p);

int length (char \*p)

```
int i;  
for (i=0; *(p+i) != '\0'; i++);  
return (i);
```

}

char \* reverse  
main()

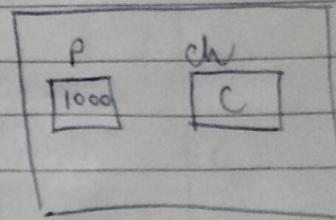
COMPUTER

{ printf ("%s", reverse ("computer"));  
getch ()

- Write a function to reverse a string  
char block ka address return Kaise  
char \* reverse (char \* p)

PAGE NO. icon  
DATE: / /

```
int l, i;  
char * p;  
for (l = 0; *(p + l) != '\0'; l++);  
for (i = 0; i < l/2; i++)  
{  
    *(p + i) = *(p + l - i - 1);  
    *(p + l - i - 1) = ch;  
}
```



## POINTERS AND STRUCTURE IN C LANGUAGE

Ques What is structure?

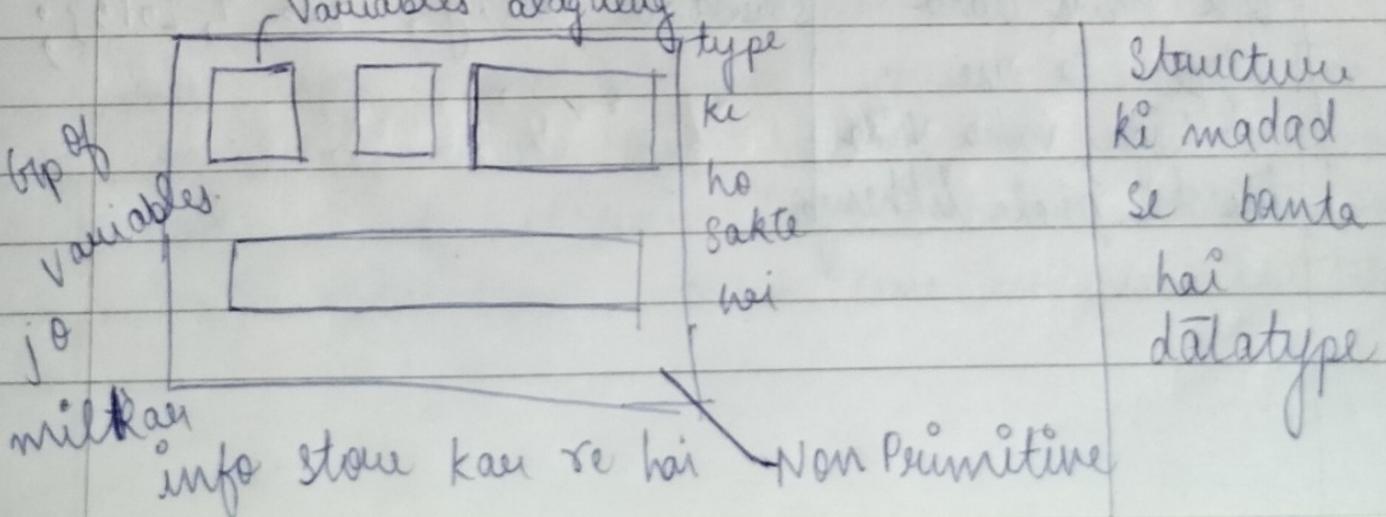
- Ans) • Structure is a way to group variables  
• " is a collection of dissimilar elements  
• Defining structure means creating new data type

Primitive = 

Variable
----------

Variable me sirf ek hi value store ho sakte hai

Variables alag alag



## DEFINING A STRUCTURE

struct tag {  
 {  
 global {  
 definition int d,m,y;  
 };

eg) struct date {  
 Ye ek data type ban  
 gaya hai

struct student

{  
 int rollno;  
 char name[20];  
 int age;}

- No memory is consumed for definition of structure
- Memory data type ko ni variable ko milti hai.

void main()	struct date
local { struct date definition { int d,m,y; };	{ int d,m,y; } & d2, d3; void main () {

ye d2, d3,  
 today sab  
 6 byte memory  
 lega.

Rule Taha jaha struct  
 & bana data  
 type ka use  
 hoga, vaha vaha  
 struct phle likhenge

float x;  
 struct date today = {26, 7, 2015};

today

d	m	y
26	7	2020

PAGE NO. 29  
DATE: / /

day, month, year ki apni alag se identity ni hai  
unko use karna hai to today likh kar. (dot)  
lagana padega.

d,

e.g.) struct date

{

int d, m, y;

}

void main ()

{

struct date today; d, ;

today. d = 26 ;

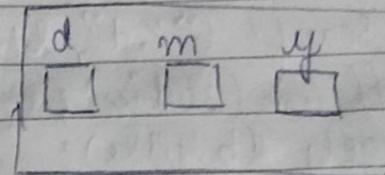
today. m = 7 ;

today. y = 2020 ;

d1. d = today. d;

d1. m = today. m;

d1. y = today. y;



or

d1 = today;

Agar user se input  
kaeani hai values

printf("Enter today's date");  
 scanf("%d/%m/%Y", &d1.d, &d1.m,  
 &d1.y);  
 getch();

Jaise aapko book ki info store karni hai, int  
 chaur vagans me surf ek ek info hi store ho  
 paigah na ki puri isliye hum static use  
 kar sakte hai. Bahut sare variables ke liye

```
#include <stdio.h>
struct book
{
    int bookid;
    char title[20];
    float price;
}
```

PAGE NO. icon  
DATE: / /

```
}*
```

```
struct book input()
```

```
{ struct book b;
```

```
printf ("Enter bookid, title and price");
```

```
scanf ("%d", &b.bookid);
```

```
fflush (stdin);
```

```
gets (b.title);
```

```
scanf ("%f", &b.price);
```

```
return (b);
```

```
}
```

```
void display (struct book b)
```

```
{
```

```
printf ("%d, %s, %f", b.bookid, b.title, b.price);
```

```
}
```

```
void main ()
```

```
{
```

```
struct book b1;
```

```
clrscr();
```

```
b1 = input();
```

```
display (b1);
```

```
getch();
```

```
}
```

## UNION IN C LANGUAGE

- Union is similar to structure, except it allows you to define variables that share storage space
- Defining union means creating new data type

Memory efficient hai

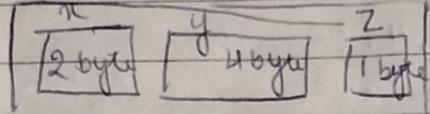
PAGE NO.	icon
DATE	/ /

## DIFF B/W UNION AND STRUCT

Memory ka diff hota hai mainly.

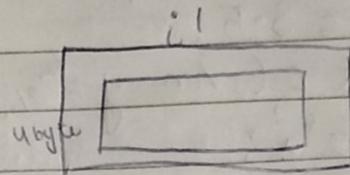
struct item

```
int x;  
float y;  
char z;  
};  
struct item i1;
```



union Item

```
{  
    int x;  
    float y;  
    char z;  
};  
union Item i1;
```



Yaha par jo variable sabse jayada hai to usko utni memory milegi

Paaantu apko teeno variables data ki ek saath jaani hoi to aap struct hi banaiye.

Eg Hame student ki info store karni hoi uski class, naam aur roll no to name in teeno ko memory deni padegi to struct

Lekin jab hame user se value input karani hoi to hame nahi pata vo kaunsi vali value dega int, char ya koi si bhi to hum union use karenge. Usse hum memory ki bachat karenge.

(e) union Item

```
int x, float y, char z;
}
void main()
{
```

union Item it;

```
i. x = 5;
```

```
printf("%d", i. x);
```

```
i. y = 3.5;
```

```
printf("%f", i. y);
```

```
i. z = 'a';
```

```
printf("%c", i. z);
```

```
getch();
```

Harne se pass 1 byte  
memory mili ho bar  
1 byte ho kya  
vo ek ek karte  
values store kar  
sa hai

x me kuch raha  
vo thi usi block  
me

Union members are accessed in same manner  
as we access structure member.

Union ka use usually low level programming  
me hoti hai high level me usually  
struct use karte hai.

## \* DYNAMIC Memory Allocation

SMA= Static Memory Allocation

DMA= Dynamic

Declaration statement ke jaise jo variable  
banate hai, vo SMA ka eg ho

A se banne wale variables ka usually naam bhi rakha jata hai.

PAGE NO. 30  
DATE: / /

eg) main()

```
{  
    int a; // SVA  
    float b; // SVA  
    int n[5] // SVA  
    - - -  
}
```

Taise hum call hoga phir  
for a ko memory mil  
jaiga. Jaise hi main  
ka kaam khatam, a destroy ho jaiga. SVA me  
ye ni kar sakte ki a phle hi destroy  
ho jai.

Static memory kyu kaha  
hai kyaki compile ke  
time par variable ko  
kitni memory milne vali  
hai vo fix ho jata hai,  
usko phir kya change ni  
kar sakte. Run ke time  
par kitni memory milgi  
vo fix ho jata hai

- DMA me hum ke time par hi decide kiya jaiga  
ki kitni memory milgi. Compiler ko  
phle se pata ni hogya ki variables ko kitni  
memory milgi.

Now we have two methods to form variables  
one by SVA and other by DMA

SVA me pura declare karke phir variables  
ka naam likhna padta hai ya phir DMA  
me variables ke naam ni hoti bas address  
hoti hai. Aise address ki madad se hi hum  
uska access kar pate hai.

Q Ab DMA kab use karne aur kab SVA?

Ans Humne jaise use se nos enter karane hai,  
agar humne phle se pata hai ki usse

if 10 nos enter karunga then we use SNA.  
agar name ni pata ki user kire nahi enter  
karunga 50, 100 ya 5000 tab hum PAGE NO.  
Karunge DATE: DMA ka use

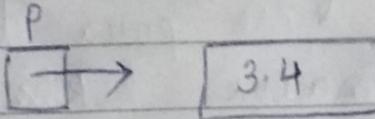
## DMA

- malloc()
- calloc()
- realloc()
- free()

malloc() function ko hum call karunge. It's not a declaration statement, its an action statement. Compiler ko samaj ni jaiga malloc(). Run pke time par pata chal jaiga kitne variables hai, aur unko kitni memory chahiye hogi.

malloc(6) = ka matlab humne malloc ko bola ki 6 byte ka memory block bana le. Agar malloc(4) bola to 4 bytes ka memory block bana lega. malloc me hum variable ka data type ni bata rahe hai, hum sledha seedha uska size bata rahe hai.

e.g) main()



float \* p;

if ()

\* p = malloc(4)

\* p = 3.4

Malloc ka written type kya hai? Void.

malloc jo address return karta hai, vo int type

kahya float type ka nota name ni pata hai.

malloc void ka nota

PAGE NO. 31 | DATE: / /

Malloc kya return kar raha hai void type ka lekin p kya hai float type ka kaide se kya kaha chahiye ki agar pointer float ka hai to ~~float~~

float ka hi address store hona chahiye. Malloc to void return karega na. Jo garbage ho jaigi to apan kya karenge? To hum karenge typecast. Typecast ka matlab hai jo value malloc return kar raha hai, uska ninte change kar re hain uska type jo hai usko change kar re hain

p = (float \*) malloc(4)

↳ float block ka address hogा jo malloc return karlega.

### - Malloc() VS calloc()

~~malloc(4) me ek argument pass karte hain. calloc(5, 2); me do argument pass karte hain~~

int \*p;

p = calloc(5, 2);

Hay block 2-2 byte ~~ka~~ hai

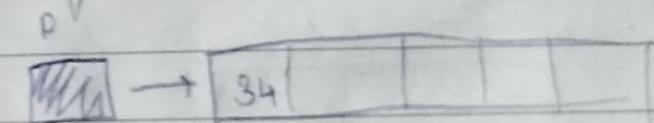
Malloc()

calloc()

① ki maddad se banta hai  
ek block

Ki maddad se ek se jyada  
blocks bante hain (array)

② ki maddad se banne valo  
ke andar garbage value  
hoti hai



$$*(p+0) = 34 ;$$

$$*(p+1) = 55 ;$$

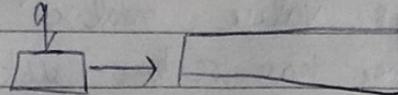
par wo malloc ya calloc ki maddad se jo banane wale blocks  
agar bad me uske size change karne ki jarurat pade to

realloc()

void\* realloc(void\* block, int size);

double\* q;

q = realloc(pte, 8);



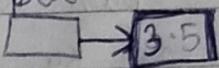
PAGE NO. icon

DATE: / /

float\* pte;

pte = malloc(4);

pte



\* pte = 3.5

free() ka kaam hai memory ulesse karna.

Eg) main()

void fun()

int x; // local variable SMA  
int \*p;  
p = malloc(2);

free(p);

p kabhi bhi SMA se bane variables ki memory release ni karta. Vo sif DSA se bane variables ki memory release karta hai.

Memory leak ho jai.

a ka kam hote hi memory release ho jaigi block khatam hoti hai.

Lekin malloc ka block khatam hoti baad bhi memory release ni hoti.

## ENUMERATORS IN C LANGUAGE

It gives an opportunity to invent own data type and define what values the variables of this data type can take.

enum month

{

jan, feb, mar, apr, may, jun, jul, aug, sep, oct,  
nov, dec

}

↑  
Ye variables mî hî

main() {

enum month m<sub>1</sub>, m<sub>2</sub>, m<sub>3</sub>

}

variables banai hî month-type ke

→ Inne 12 values store kar sakte hî.  
jo hame phle hi bata di hî.

- Each value on the list of permissible values corresponding to an integer, starting with 0. In the example, jan is stored as 0, feb is stored as 1 ..., dec is stored as 11.

Agar hum seedha m<sub>1</sub> = 3 likh dete to  
bahut confusion ho jaegi 3 & march wo  
sakte hî ya april bhi wo sakta hî

- Change the default value of enum ~~value~~ elements.

enum month

{

jan=1, feb, mar, apr, may, jun, jul, aug,  
sep, oct, nov, dec

Another eg  
num ~~bool~~  
E  
3; false, true

You can write any program in C language without help of enumerations but, enumerations helps in writing clear codes and simplify program long

Check a number is even or odd?

Ans.

enum boolean isEven (int n)

```
enum boolean  
{  
    false, true  
};
```

if ( $x \% 2 == 0$ )  
return (true);

else return (false)

3

```
void main()
```

int m;

enum boolean result;

classical ( );

printf ("Enter a number");

'scarf' (66% d ?? & m);

result = isEven(n);

if (result == true)

sum of ("even numbers")

else

print f("Odd number")

getch()

## typedef

① typedef is a keyword.

② We can use typedef to give new name to a type

typedef int LENGTH;

③ Now you can use LENGTH as a data type which is just same as int

LENGTH x, y;

④ By convention, uppercase letters are used for these definitions to remind the user that the name type is really a symbolic abbreviation but we can use lowercase.

typedef int length;

- typedef a structure

struct Student

{  
int roll no;  
char name[20];  
int age;  
};

{  
eg) struct student

{  
int roll no;  
char name[20];  
int age;  
};

{  
void main()

{  
struct student s1;

} OR

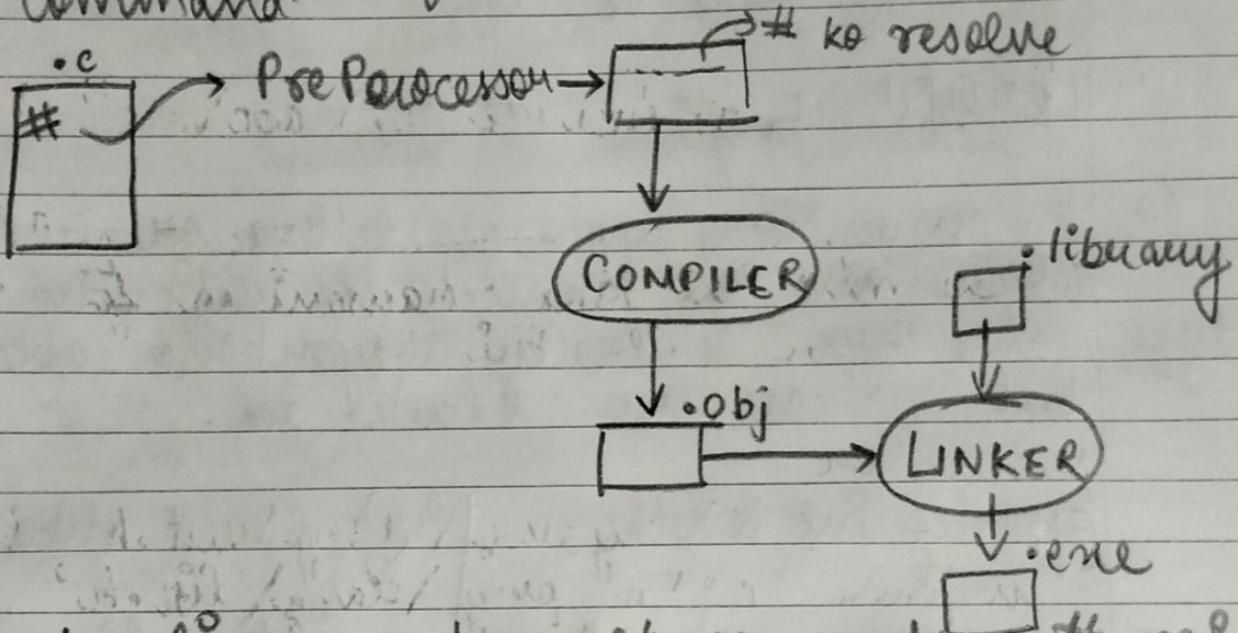
typedef struct Student

{  
int roll no;  
char name[20];  
int age;  
};  
STUDENT;  
void main();

{  
STUDENT s1;  
s1. roll no = 34;  
strcpy(s1. name, "SARUBH");  
s1. age = 20;

## Pre processor?

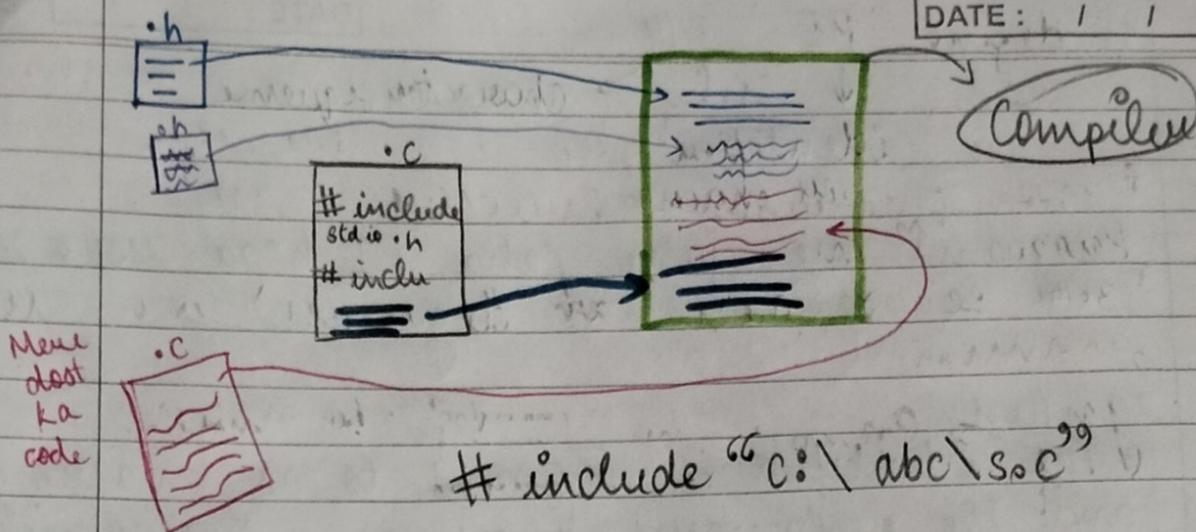
- Pre processor is a program which performs b...  
DATE: / /
- Pre processor only notices # started statement
- # is c/d pre processor directive
- each preprocessor directive must be on its own line
- The word after # is called pre processor command.



- Each line me hum ek se jyada # ni laga sakte.

### - # include

- include is one of the most popular preprocessor command
- It can be used include any file content to your source file.
- `# include < file name >`
- `# include " file name "`



Humne yaha par angular bracket ni lagai coz  
jaewi ni hai jaha hamari .h file takhi  
he vahi .c file bhi he.

### Remember

- #include <c:\myprog\clang\list.h> is wrong
- # include "c:\myprog\clang\list.h" is correct.
- Inclusion of another source file
- You can write your source code in multiple .c files
- You can include all these source files in same way as you include header file.
- **# define**
- The # define directive defines an identifier and a character sequence (set of characters) that will be substituted for identifier each time it is encountered in source file.

# define PI  $\downarrow$  3.14  $\rightarrow$  character sequence  
identifier  
or macro

Program me jaha jaha PI hoga usko replace  
3.14 se replace kar dega.

### Syntax

- The identifier is referred to as macro name and the replacement process as macro replacement
- # define macro name char sequence

Ex) # define PI 3.14  
int main()

{

int r;

float a;

printf ("Enter radius of a circle");

scanf ("%d", &r);

a = PI \* r \* r;

printf ("Area of circle is %f", a)

### - Defining macro as a function

Ex) # define SUM (a,b) a+b  
int main()

{

printf ("Sum of 3 and 4 is %d", SUM(3,4));

getch();

SUM (a, b)

SUM (3, 4)

3+4

# define PRODUCT (a, b)  $a^*b$

int main()

{

printf ("Product of 3 and 4 is %.d", PRODUCT(3,4));

getch()

}

PAGE NO. 33  
DATE: / /

Eg)

# define PRODUCT (a, b)  $a^*b$

int main()

{

printf ("Product of 3+2 and 4-6 is %.d", PRODUCT

(3+2, 4-6));

getch()

}

Product (3<sup>a</sup>+2, 4<sup>b</sup>-6)

a = 5 X

3+2 \* 4-6

a = 3+2 ✓

3+8-6

5 ✓

Eg)

# define square (a)  $a^*a$

main()

{

int s = square (5);

printf ("Square is %.d", s);

}

-

# undef

It is used to undefined Macros

\*

Eg

#define CLOSE 0

...

...

# undef CLOSE

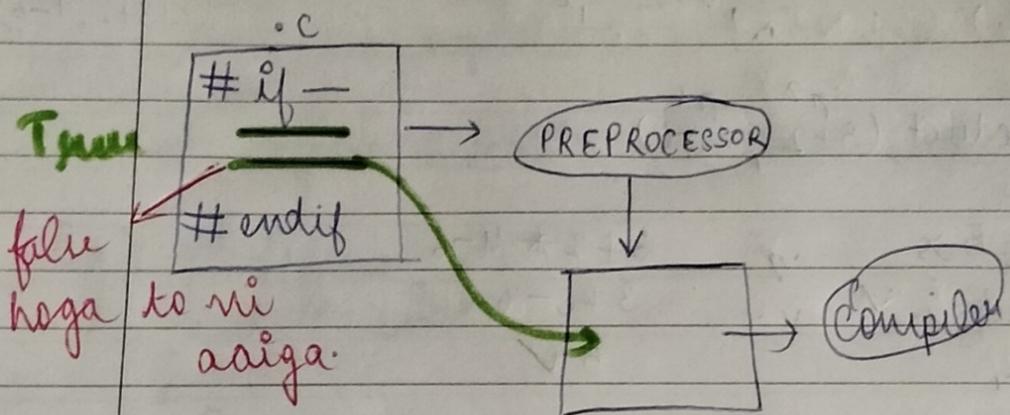
Pre Processor Commands

- # if, # else, # elif, # endif
- # ifdef ; # ifndef
- ##

PAGE NO. icon  
DATE: / /

# if, # else, # elif, # endif

- As a file is being compiled, you can use these commands to cause certain lines of code to be included or not included (for compilation)



leg) # define COUNTRY India  
ye false hogi tabhi # if COUNTRY == pakistan  
is par aiga. printf (" Pakistani Rupees");  
# elif COUNTRY == bangladesh  
printf (" Taka");  
# elif COUNTRY == Nepal  
printf (" Nepali Rupees");  
# else  
printf (" Indian Rupees");  
# endif  
ye properly likhkar hi khatam karne hain.

} ye condition  
} false hai ye  
compile ni  
hoga.

#ifdef, #ifndef  
#ifndef macro

- If the macro has been defined by a

#define statement, then the code immediately following the command will be compiled.

#ifndef

- If the macro has not been defined by a #define statement, then the code immediately following the command will be compiled.

Eg) #define COUNTRY "India"

#ifdef COUNTRY

printf ("%s is a great country", COUNTRY);

#endif

#ifndef COUNTRY

printf ("I Love my Nation");

#endif

##

- The operator ## is used to ~~defeat~~ with the #define macro.

- Using ## concatenates what's before the ## with what's after it.

#define ACTION(a,b) a##b + a\*b  
main()

{

printf ("%d", ACTION(3,4));

}

$$34 + (3 \times 4)$$

$$34 + 12 = 46$$