

- why C language is imp?
- Ondde 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 hui ek cheez ko 0 ya 1 ki form me code karne ka taaleka nikal rakhna 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 kon banata jata hai. Agar N pole niche banata hai to 1. N pole upper banata hai to 0. Magnetic material me kai saari places hoti hai uss puri hard disk me apni bahut sare 0 aur 1 gare hote hai.

16/8/2024 Filobots
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 ke sare kaame ke liye ek capacitor

Haard
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 ka milakar ek device banainge to banega 16 flip flop a ek register. 16 o, 1 Voltale

- Hardware -
Monitor, keyboard, scanner, etc. Sab Hardware nota hai. Ye ek physical cheez hoti hai jisko 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 kis se pata kar jata hai taaki ka data rakha huwa hai

jpg form me hai to ^{0,1} isse image ki pixels ki coding mani jati hai

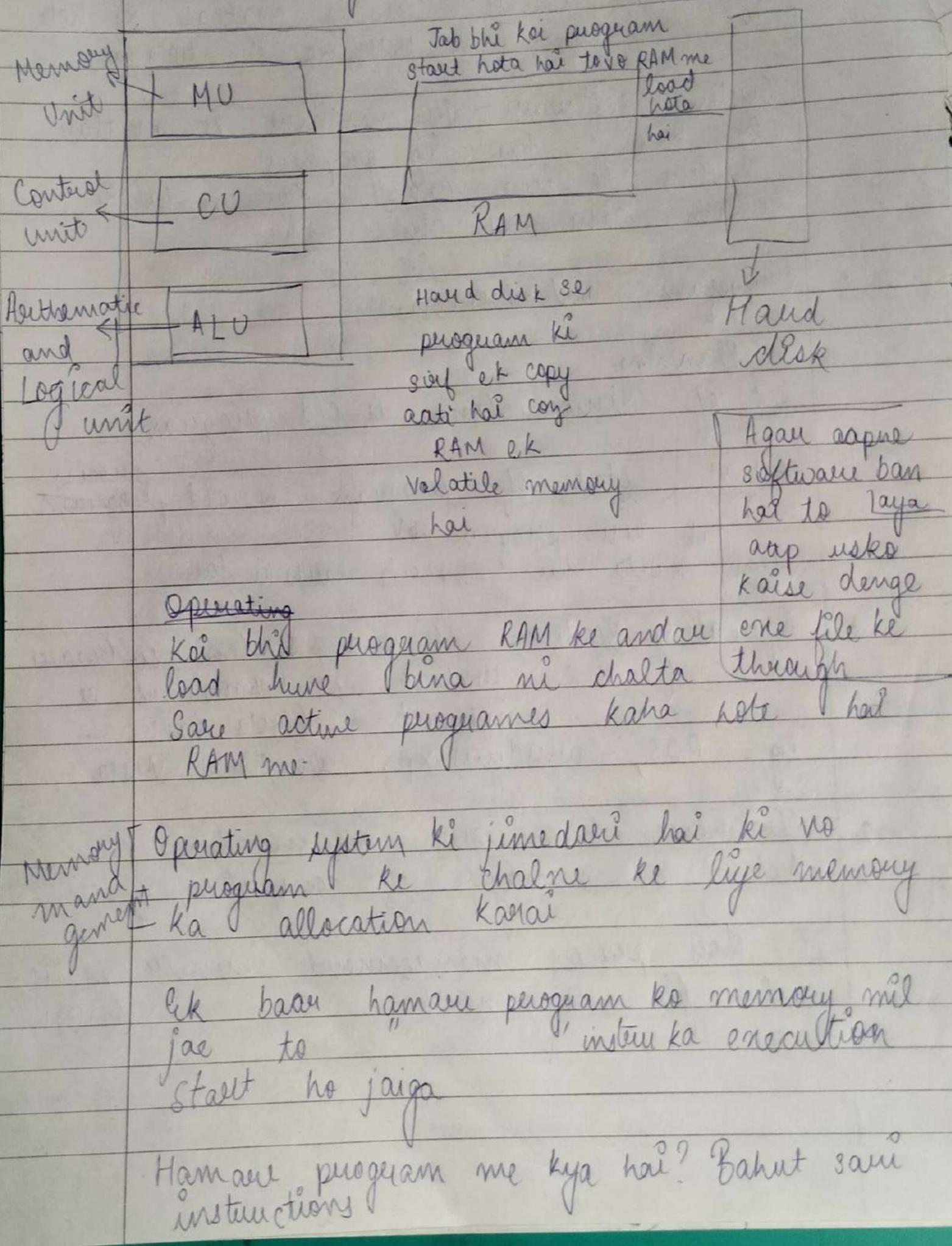
txt = koi character ya koi symbol
software exe = executable file Special data iske
0,1 ka matlab comp ke liye muidesh

ki comp ko kaerna kya hai
exe file software.

PAGE NO. 5
DATE: / /

- Application software - User ke matlab ke note hai. calculator. Jisse user ko " hota hai
- System software = Jo machine ke matlab ke note hai. e.g. graphics system ke liye kaam karne ke liye 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
e.g. = DOS, windows xp, windows vista,
- 1 Interface b/w user and machine
- 2 Act as manager of computer system
- 3 It does process management, memory management and file management

Understanding execution of a program



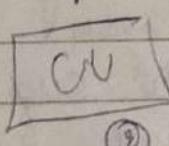
16 bit ke architecture ki hisab se first 0,1 hai jo hamara first instruction hai. Agle 16, 0 and 1 kya hai doosre instructions esse kai same instructions hamare program me likhi hui hain.

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

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

1. MU \rightarrow Set of registers

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



②

2. CU \rightarrow Circuit that reads instructions and decode it

Ye samajta hai instructions ko ki kya karna hai?

- ③ CU, ALU ke 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.

3. ALU \rightarrow Responsible for all arithmetical and logical calculations

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

• SOFTWARE DEVELOPMENT IN C.

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

Eg apan ko banani hai ek file Sum.exe

Apni file ko banani me ye dikkat hai ki ye operating system dependent language

| |
|--------|
| 010101 |
| 011100 |
| 011100 |
| 000111 |

hai matlab agar ye file doc operating system ke samajne layak hai to kisi aur operating system ko samaj me nahi aaiga. To agar mai is file ko banata bhi hu to suff ye doc operating system par chalgi 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 is 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 padgi Isse aasan tarika hai ki hum kya seekhe c language.

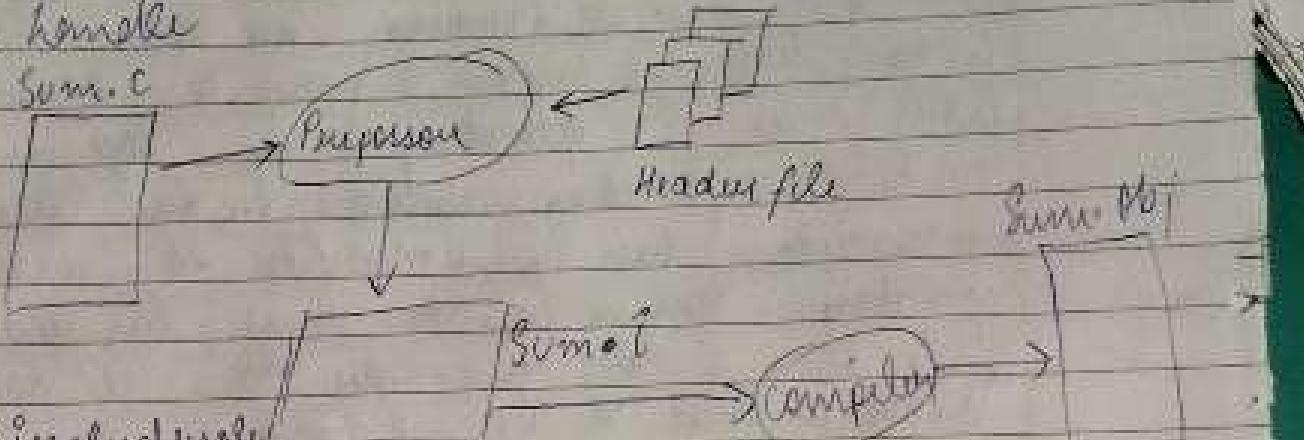
To apne kya karenge c language me coding
 karne ke liye to iss file ko kya bolengi source
 file Iss file ka extension kya hogi .c
 Ye file me c language me apne ne kya
 likha hai jo name comp se karanा hai
 Lekin ye operating system software ni hai
 kyuki operating system .exe file ko
 samajta hai, .c 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 karne. 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 rahi
 hain unko handle karne ka kaam iss software
 ka nota hai

Hamne program me jitne bhi hydrofiles
 ka zikr hota hai, unne hydrofiles 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 hydrofiles 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

yaha se hath chuke hai aur dosen statement
 yaha se chuke hai? basically preprocessing
 se start hain uski statements ke aache
 se honenge

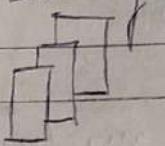
Source



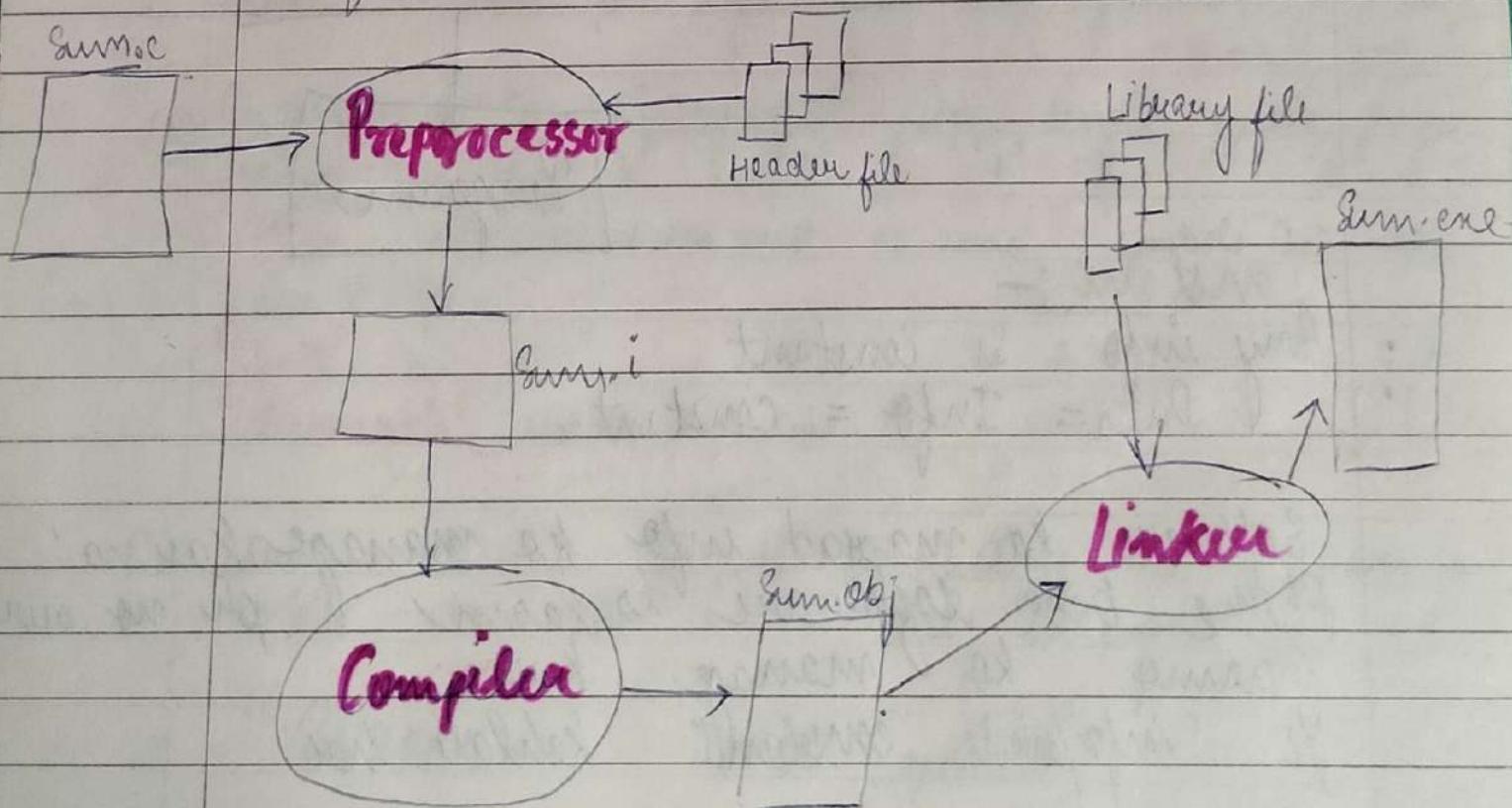
/ includes

Eg. #include std::iostream
 Yeh ek command hai,
 Jiska matlab hai ki Header file std::i
 ki andar jo bhi content hai usko progra
 m se jod do to preprocessing of std::i wali line
 ke hatkar aur header file std::i me jo bhi
 likha hai usko program me sharif kau
 dega. Is tarah us code includes ho jayega
 jo header file me likha hai. Jo file bani tab
 usko hum compiler software ki madad
 se hum kaurae hain translate. Compiler hamara
 program ko translate karega us mai file bana
 dega jo particular operating system ki
 hisab se bani hoga. Compiler along along
 operating system ki hisab se alag-alag
 aata hai. Koi compiler doge jo hisab
 se convert karega to koi lines ke terms
 Compiler me jo file bana hai, uska
 extension hain .obj. Jo main humne
 apne file ka rakhya tha, compiler thi
 raha main rakh jao.

Apne program me use banne sare cheze 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 karna ke liye kya use karvengi? Liken liken done ka file lik karke ek naya file bana dega. Aur iss taarak se hamara software bana jaiga.



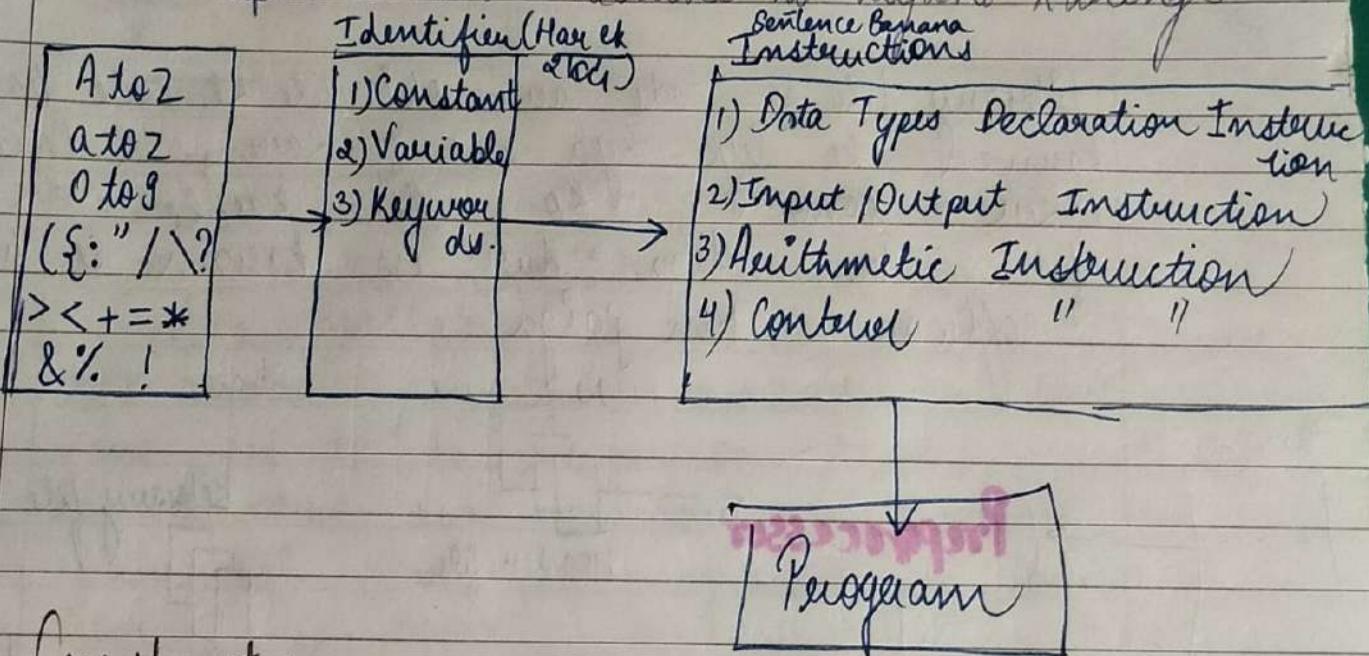
.c file ko software ni hain une file soft ka hain

यदि आप अपने प्रोजेक्ट की डिको में file sum.exe कही तो उसे

Compiler software ki jaaniat nahi hogi. Humne 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 nichali machine ko vayakt karenge



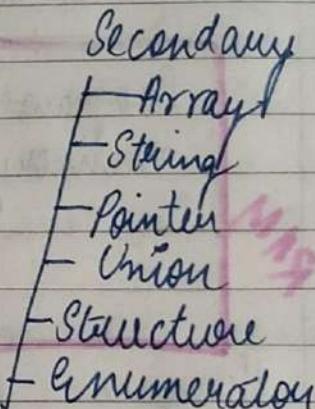
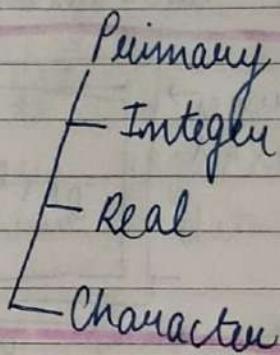
Constant :-

- Any info is constant
- Data = Info = constant

Software ka magad info ko manage karna
Phone book software ka kaise ph no aur
naams ko manage karna
Ye info/data is constant kehlati hai.

Types of Constants-

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



1. Integer constant - ek no. hai jo aur esa 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

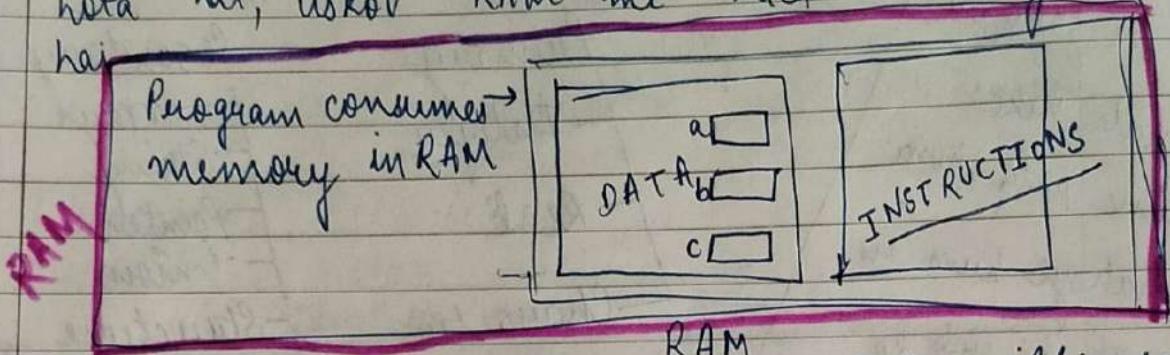
3. Character = koi bhi symbol jisse, single, quote, laga hua ho. eg 'a', 'B', '+', ',', '2', laga '-3' Not a character const as isme 2 symbols hai - bhi aur 3 bhi.

'3.4' " " " " " 3 " " " 3, . 4
'Samarth' Not a character

4. String = ek se jyada characters ko mil ake 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 ^{RAM} memory milti hai, ek instructions ko rakhne ke liye aur doosaar data rakhna jaiga.

Kisi bhi program ke do ahein 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 nos ko add karna hai, to appko 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 appko 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 mane 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
- ~~Var~~ 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~~ or Sochiye, haar ek shabd ek identifier hai. Compiler hamare program ke padhkar trans karega aur obj file banaiga. Compiler late ko hamare program ka haar ek ~~ETC~~ Samaj me aara chahiye tabhi vo usko translate kar paiga. Agar kuch ~~ETC~~ esse istamal kiye hai jaise variable ka naam, compiler unko digest nahi karta. Isliye aap haar ~~ETC~~ or matlab khud batate hai. Lekin ~~ETC~~ esse

bhi hai, jinka matlab aapko compiler ko batane
 ki jaannat ni hoti. ~~kuch~~ Unka matlab
unke compiler ko phle se pata hota hai
 esse ~~& TO~~ * Note hai keywords.

~~Key~~^{words} = 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 | | |
| const | for | return | typedef | | sakte |
| do | float | short | union | | |

DATA TYPE DECLARATION INSTRUCTION in C

Instruction -

Jaise hum statements bolte hai, waise hum program
 ing me unhe instructions bolte hai.
 Hindi me hum unhe ~~data~~ Bolte hai

~~too~~

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

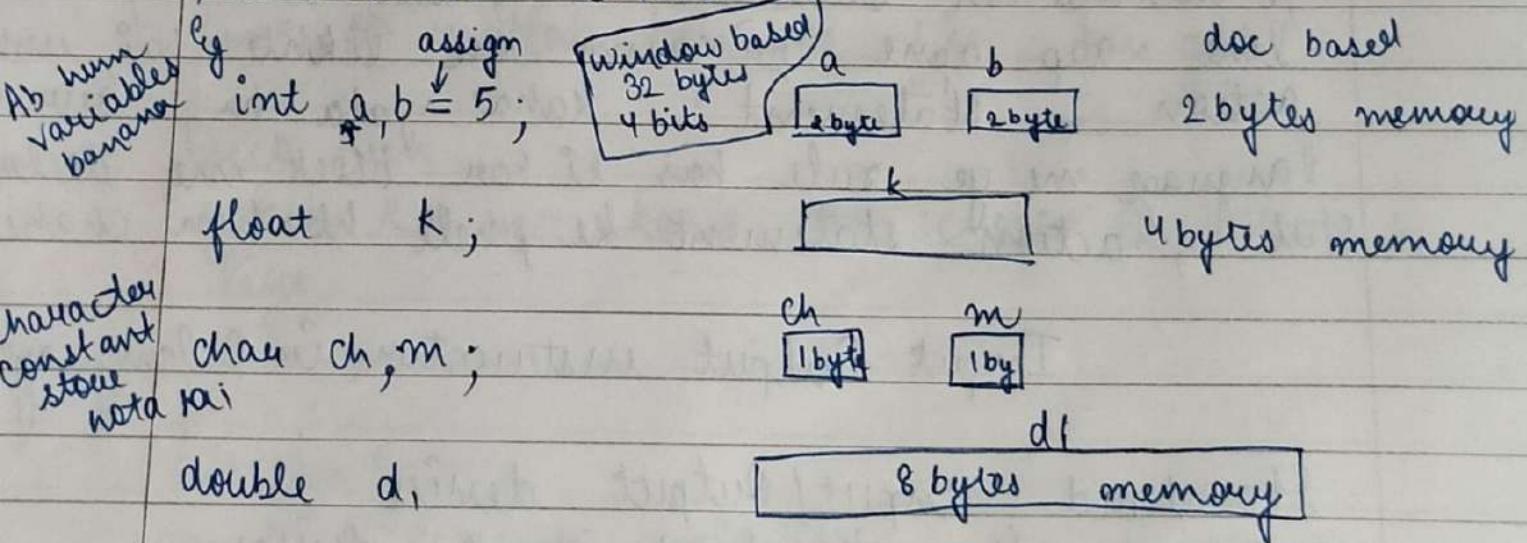
- Data type Declaration Instruction
 - Input Output Instruction
 - Arithmetic " "
 - Control "

DATA TYPE

- int
- char
- float
- double
- void

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

DECLARATION STATEMENTS -



Jab compiler apne program me essa dekhega to vo samaj jaiga ki ~~is~~ ye data type declaration instruction hai jiska phela ~~is~~ ~~is~~ Data Type hi hena 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 inspred table value hai, garbage value. Hum declaration ke baad bhi value initialise kar sakte hai. To jaenu 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 rai 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 declare 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 apke program me jiske chalne par screen par monitor par koi message print hota hai. Screen par kuch print karana hi output rehlaiga. Eme 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
 naam de rakhा hai
 printf()

PAGE NO. 10
DATE: / /

Two types of messages

- Printing text as it is

- Printing value of expression or value of variable

main()

→ output instruction
Ka kaam
ek chalak
keyword

printf("Saumub Shukla");

getch();
se lena
ge har

Input instruction

Compile

Compile

Link

Warning:

Link ka matlab hai one file ban ga

Click Run

getch ko use karte hai screen ko rokne ke
liye

{

clrscr();

printf ("Saumub Shukla");

getch();

}

main()

{

- clrscr(); // to clear screen

printf ("Saumub");

printf (" Shukla");

getch();

}

→ command \$ g++ compile n
hoga

```
main()
{
```

```
    clrscr(); // to clear the screen
```

```
    printf ("Samabh\n");
```

```
    printf ("Shukla")
```

```
    getch();
```

In ki
vajah se
cilesa
next line
par aa jaiga

Do baar printf likhne ki jarurat ni hai

```
printf ("Samabh\n Shukla");
```

In kya kehlata hai Escape Sequences

Special symbols jo \ se

eg \n = Next line start hotel hai

\t = Space aa jati hai beeth me

\b = MANPREEKaur PEARKaur

\\" = MANPREET\KAUR

\" = MANPREET "KAUR"

\u = KAURREET

Raoekaur = 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

```
clrscr();
```

```
gotoxy(40,13);
```

```
printf ("Samabh");
```

```
getch();
```

main()
{

Declar
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 karani 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: / / Adds

main()

{

int x;

clrscr();

scanf ("%d", &x);

getch();

Keyboard se vo
value lega

Scarf bhi eco
kanta hai

aur phir usko

batai gai variable me store
karna.

Keyboard se
kuch data lma
hai to scarf ki
janmat padegi

Screen
clear
abhi koi

scanf ("%d", &x);

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

main()

{

int n;

clrscr();

printf ("enter a number");

scanf ("%d", &n);

getch();

Jee vo
karna
kyo kai

main()

{

int x, y;

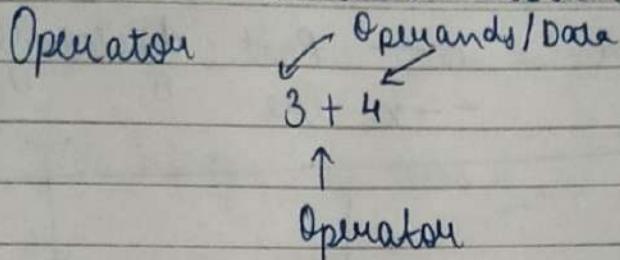
clrscr();

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 operator, is known as Arithmetic Instruction
- There is no BODMAS in C language.

e.g) $3 + 4 * 5$

35 23

main()

{

int x=3;
clrscr();

x++; // $x = x + 1$

printf("%d", x);

getch();
}

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

- Unary +, -, ++, --, sizeof()

↓↓
Sign batane wale

++ \Rightarrow increment operator

Pre increment

Run = 4

↓
++x likho

ya x++ same

baat

Post
increment

{
int x=3;

x ++;

printf("%d", x);

printf("%d", x);

Run = 45

↑ Dosseu baar

ek baar } add kya
phle }

Post increment ki priority sabse kam hoti hai

Pre increment " " "

pragatisha 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()

L5, P3

Bytes

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

int x;

x = size of (float);

printf ("%d", x)

getch();

~~#~~ main()

(double)

(int)

8
4
(char)

Run 4

int x, y, z;
^{Integer}
^{conv}
^{Real}
x = size of (34);
y = size of (3.56);
z = size of ('a');
printf ("%d %d %d",
 x, y, z);
getch();

eg {

int x, y;

float k;

double d1;

char ch;

clearscr();

x = size of (y);

printf ("%d", x);

getch();

Run 2

y ek variable
hai, jiska type
int hai

282

Double ka size jayada
nota aachhi se decimal ko

(K) 4

2 = size of ('a'); // ASCII

↑
character constant

Hence each character constant has code value 1
a ki ASCII value = 97

0.4255
Date

PAGE NO. 12

DATE

97 integer to ans 2 Comp 4

- Arithmetic Operations

Priority

* / %
+ -

Priority bracket hain those ki
left to right solve kipa jiga

3 + 4 - 7 To phle + karne me -



Priority same

main()

{

int x;

x = 3 + 4;

printf ("%d", x);

getch();

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 operat
perform karne ke ans
lise

Modular Op

integer hi aaiga

x = 37.4 remainder x = 1/5

Run = 2

Run 3 data hai x = 1/6

Run = 1

x = 3.0/6

Real Result

x = 137.4 Run 1

x = 237.5 Run 3

ek bhi real hai to result real

x = 251.5 Run 0

aaiga

some integer hai tabhi
result integer alga

3/4 0

3.0/4 0.75

3/4.0 0.75

3.0/4.0 0.75

visibility kise check hogi?

odisha paper after se

BITWISE OPERATORS

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

Exclusive OR

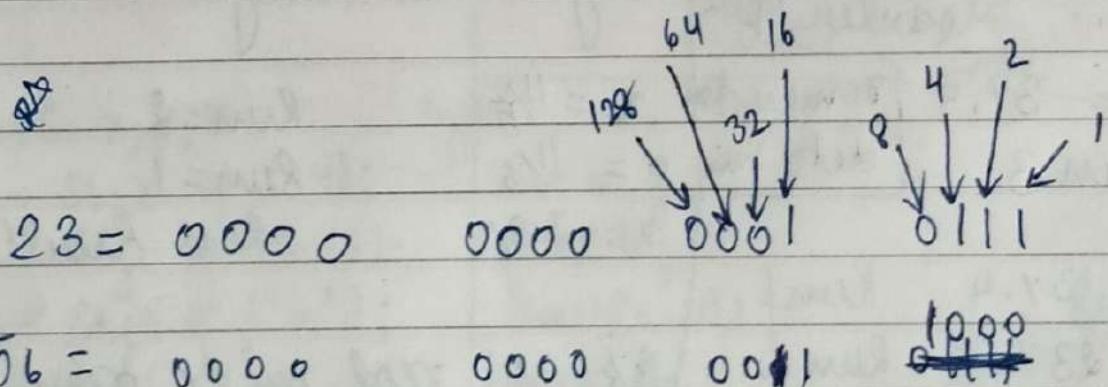
Surf
Dain / pe
kaam karte
hai

| & 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 0011 1111 = 63

Ig)

call
kiya ←
main()
{
 main()
 {
 add();
 getch();
 }
 void add()
{

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

define
kiya hai

scanf

Lekin printf, ko call karte samay parenth
empty ni hoi to ye takes some thing is
nature ka hoi.

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

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

Return kaunse ka matlab add function ko fisre call
kiya hoi, usko value layana

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

koi function agar predefined hai to uska declaration
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
 kar dete hain like prefer karne kaun chahiye
 declaration karne ka
 Takes nothing Returns Nothing
 void main()

function void add (void); empty
 declaration

{}
 class();

add();

getch();

return;

whatever void add ()

type

{

int a,b,c;

printf("Enter two numbers");

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

c = a + b;

printf("Sum is %d", c);

- Takes something, returns nothing -

void main()

{

int x,y;

void add (int, int);

pur.

Han function
 ke bare
 variable
 hote hain

Do some functi-
 on values
 ko le sakte
 hain

Variables excep-
 ni han sakte

Takes ka maliak
 parenthesis
 me ekhi
 baat se
 sayay
 aiga aur
 return ka
 void se

→ Idher likhte to koi bhi function call kar
pata ye haq global declaration

PAGE NO. 10
DATE 10/10/10

local declaration

de
Add ko
main
solo ka

call
gaha hai
isliye uski

main ke
andar
likha hai

confuse
no sakte
no

```

void main()
{
    int x, y;           Apan me do ka
    void add (int, int); define me do van
    class();           kaise hai
    printf ("Enter 2 nos.");
    scanf ("%d, %d", &x, &y);
    add (x, y); // Actual Arguments | Call by Value
    getch();
}

```

```

void add (int x, int y) // Formal Arguments
{
    int c;
    c = x + y;
    printf ("Sum is %d", c);
}

```

Its like apan ko add karne ke liye kitni values chahiye hain do to apan ne do baar int a, int b likha. Add ke liye variables kitne chahiye hain 3

→ Apan ne idhein print f, scanf kyu nہ likha? kyu ki apne paas phle se values hain to user se magne ki koi jaannat nہ hain

x ke value a ko assign hogā hē^l
Vaise hi y " b " " " " " " " " "

TAKES NOTHING, RETURNS SOMETHING —

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

```
int s;  
class()
```

```
s = add();  
printf ("%s", s);
```

```
getch();
```

int add()
{

int a,b,c;

```
int a, b, c;  
printf ("Enter two numbers");  
scanf ("%d %d, &a, &b);
```

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

$$C = a + b$$

return (c);

3

↓
Return apan c ko mati
balki uski value ke
kam rahi hai

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

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

Return ki madad se apne ek hi value return kara sakte hain. Ek se jayada niche kyu nahi kyu ki jo value return karoge wo function call ki jagaj aye.

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

Sum ki value return karani hai to phir apni
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 chlegi.

TAKES SOMETHING, RETURNS SOMETHING

int add (int, int);

void main()

{

 int s, x, y;

 clrscr();

 printf ("Enter two numbers");

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

 s = add (x, y);

 value pass ho jaigi

 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 hain koi bhi function, kisi bhi function ko call kar sakta hai.

Eg) main()

{
int k;

Actions
Stat
ememt
} 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 kar
rahe hai.

main()

k [6] k = fun(3); ←
printf ("%d", k);

fun (int a)

a [3] if (a == 1)
return (a);

s [6] s = a + fun(a-1)
return s

fun (int a)

a [2] if (a == 1)
return (a);

s [3] s = a + fun(a-1)
return s

fun (int a)

ye fun a [1] if (a == 1)
chalta ga return (a);

s [] s = a + fun(a-1)
return s

Nahi copy RAM
me aa
jaiga

idher
return
hogi
a ki value

Iska
karun
khatam to
wki memory
release ho
jai gi.

No function

Khud ko baabat
call kar raha hai to ek pt par khatam
bhi karne hai to dead end jawaabi hai

1

Sift away ke
Leave me use
orange.

ARRAYS

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

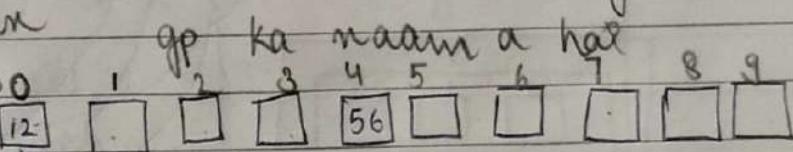
e)
g)

main () { } No of variables → 0 in age. 12

int a[10];

$$a[0] = 12;$$

$$a[4] = 56;$$



Inka individually koi naam
ni hai.

Same ek ~~ba~~ hi type ke hai

Isko access karna hai to

a [o]

```
    }  
    main()  
{
```

Avg may
be in
point

```
int a[10], i; sum = 0;
```

float avg

printf ("entre 10 nos");

```
for (i=0; i<=9; i++)
```

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

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

Aug 2000/10 p. 1

$$\text{avg} = \text{sum}/10.0$$

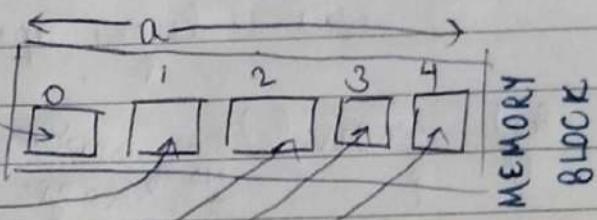
```
printf ("Average is %.d", avg);  
getch();
```

getch();

۲۹

Declaration in ARRAY

- int a[]; → Agar empty hai to compiler aa jaiga coz compiler ko pata ni chalega, kitne variable hai.
- int a[5] = 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 hain.
- 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 karega jab declare karlete huye ni initialisation kara ho.

TWO DIMENSIONAL ARRAY

PAGE NO. / /

DATE: / /

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

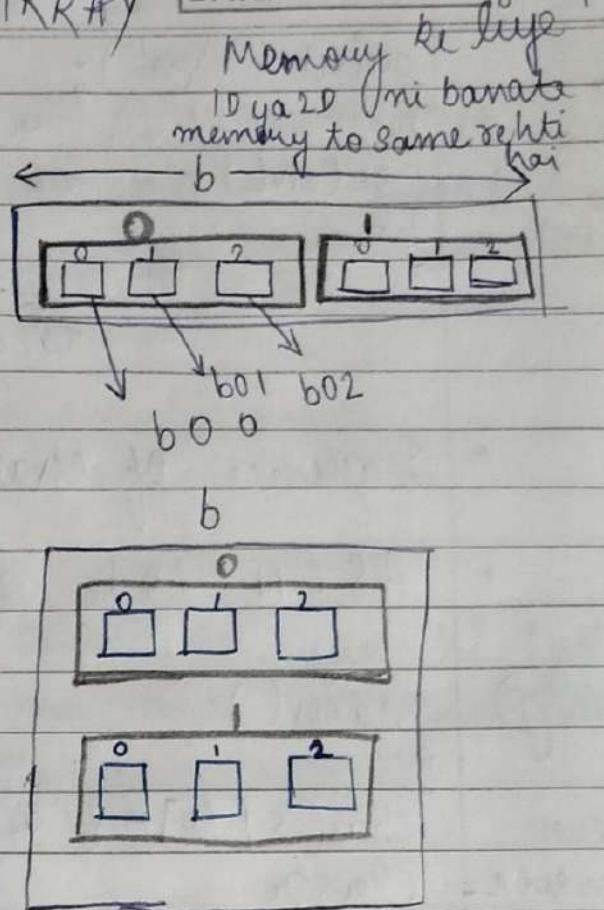
Array of arrays.

Q Why 2D Array?

Bachet same grouped data
ke liye

Eg 1 School st. 5 classes
10 students 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 [ctn]
- ASCII code of null character is 0 (zero)

Ex) main()

RUN
SAURABH -

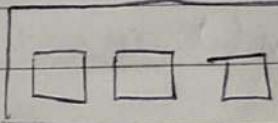
char s[10] = { 'S', 'A', 'U', 'R', 'U', 'B', 'H', '\0' };

Character

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

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

getch();



Character ko print karna

hai

→ Null character

} OR

null ki

madad se,

string me

kitne character

hain hain Jane

ki jaune ka

ni padti

main

{

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

int i;

Array ke block

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

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

getch();

Alag alag characters ko print

karna raha hai

→ Pure string ko ek baar me print

printf ("%s", s);

karna le

getch();

ek character
ko jab
examine
karna hoga

Pure
string
ko print

printf ki jagah puts bhi use kar sakte hai.
 puts sirf string ko print karne ke kaam manta hai
 puts(s); RUN = SAURABH

PAGE NO. 23

DATE: / /

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 tarike

Eg) main()
{

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

User se bolte hai naam enter karne ke liye

main()
{

char s[20];

int i;

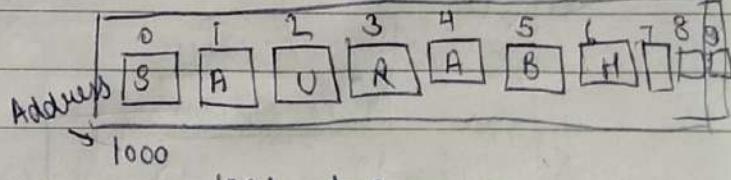
printf ("Enter your name");

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

puts (*s);

getch();

Address
of likha
hai



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

printf

s likhe ya
& s[0] ek

Scanf ek se jayada string ~~string~~ ka input ni
le sakta

scanf ke laga
data khatam baat

delimited by
No space
No enter

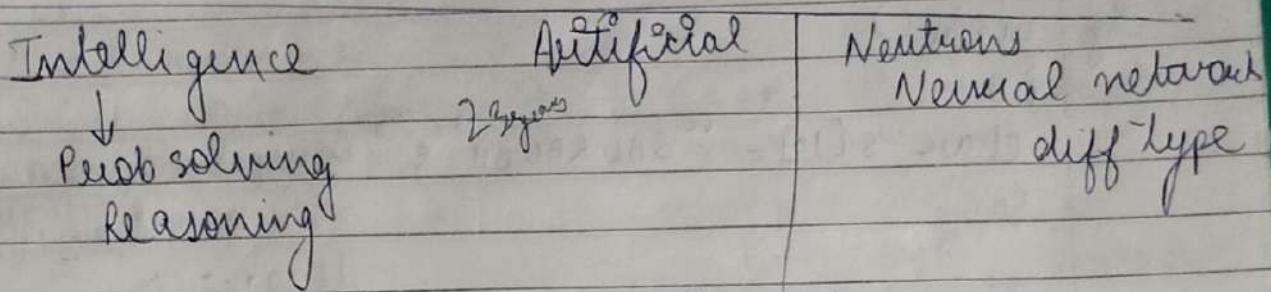
AMIT
RUN
AMIT

SAURABH_SHUKLA
Run ignore ho gaya
SAURABH

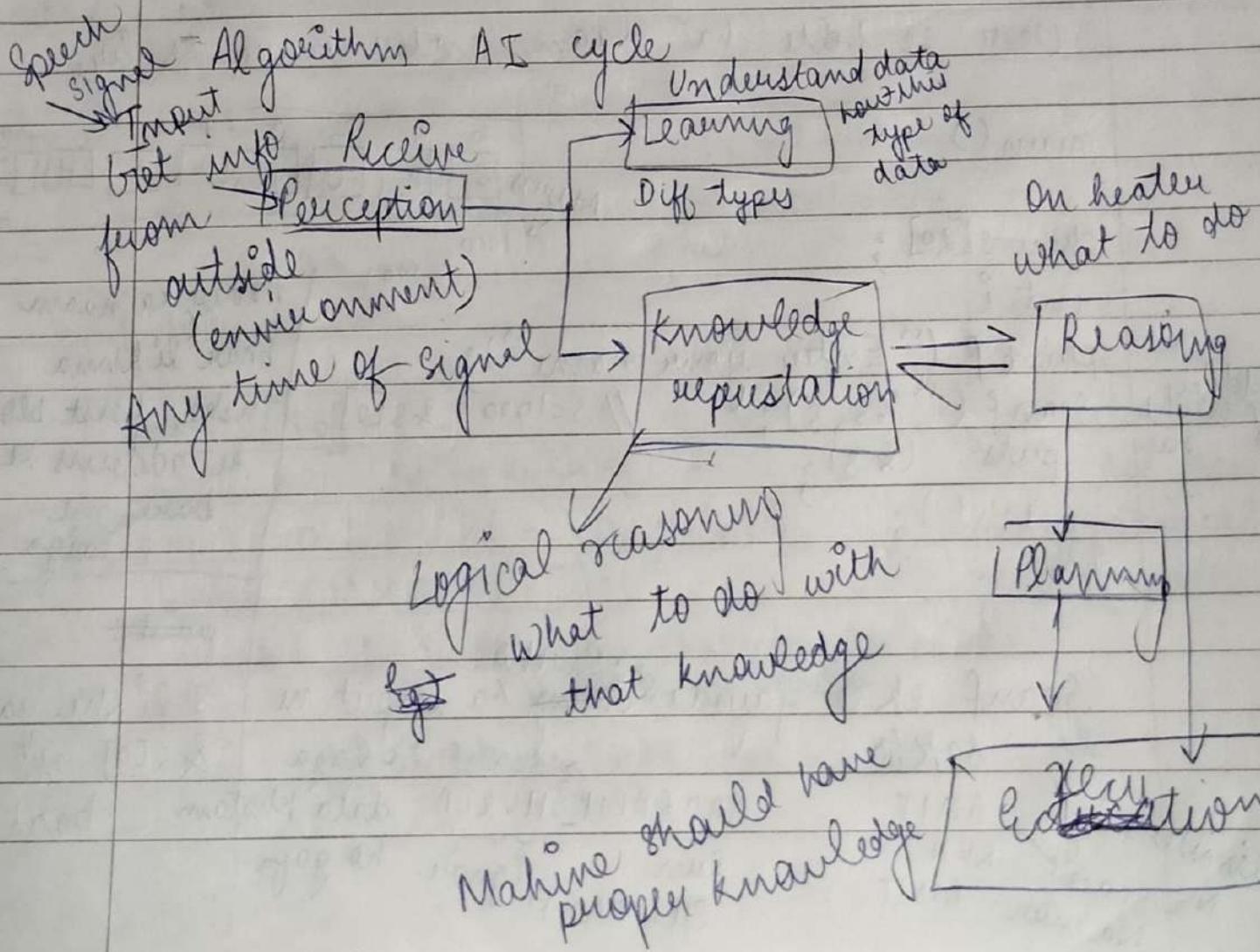
Apar scarf ki jagah gets(s) use karne wai vo
 do string ko le lega
 gets input line ka kaam karta hai !

PAGE NO. _____
 DATE _____

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



Machine should be intelligent



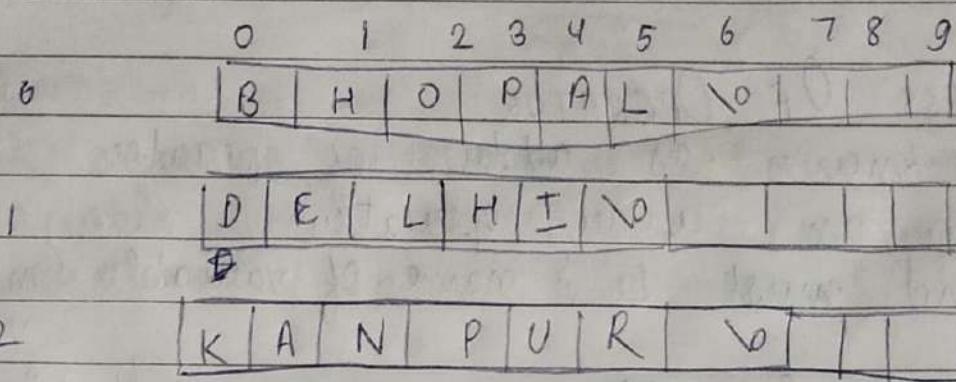
String related function

- 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])

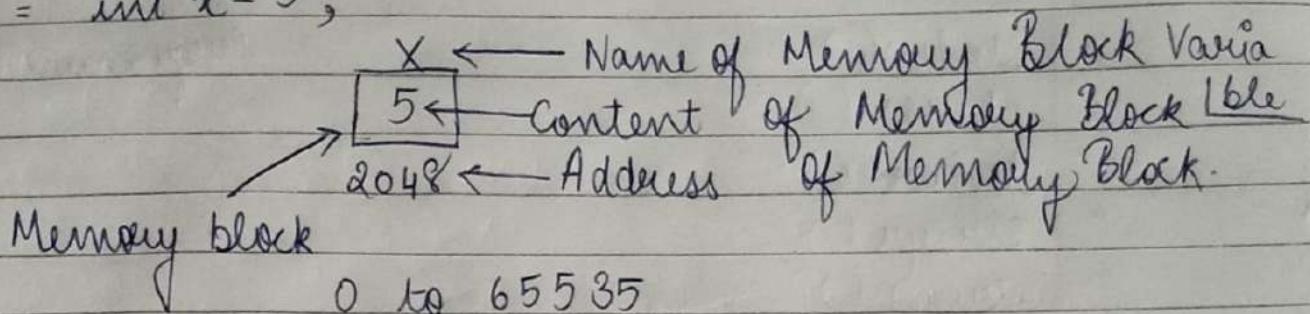
for (i=0; i<=2; i++)

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
char if not constant
- & D 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 dete hain

Eg) main()

Run

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

| |
|------|
| 5 |
| 2048 |
| 5 |

%d - 32768 to 32767

%u 0 to 65535

Ques)

```
int n=5;
&n=7;
```

| |
|------|
| x |
| 5 |
| 2048 |

error

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

Address ke right me constant n likh sakte, left me to likh sakte hai.

Eg) j=&n

| | |
|------|------|
| x | j |
| 5 | 2048 |
| 2048 | 3000 |

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

→ Yesif symbol hai

int *j Apan ne compiler ko bata diya hai ki
j special tarah 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. DATE: / /
- P** **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 %u\n", x, j);
```

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

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

cancel

| x | j |
|------|------|
| 5 | 2048 |
| 2048 | 3000 |

Asterisk aur imprecent saath
me aai to cancel kar do.

Pointer se phle * aai to **(*)** ye pura vo ban jaiga
jiski taraf apni point karna chah raha hai.

Run 5 2048

5 2048

2048.

BASE ADDRESS

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

| |
|------|
| 1000 |
|------|

| |
|---|
| a |
|---|

1000 1001

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

| |
|------|
| 2000 |
|------|

| | | | |
|------|------|------|------|
| b | | | |
| 2000 | 2001 | 2002 | 2003 |

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

| |
|------|
| 3000 |
|------|

| |
|--|
| |
|--|

3000

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

| | |
|----------|-----|
| PAGE NO. | 26 |
| DATE: | / / |

Pointer me sirf base address hi jata hai.

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

j aks point kar saka hain.
 k bks " "
 l c " "

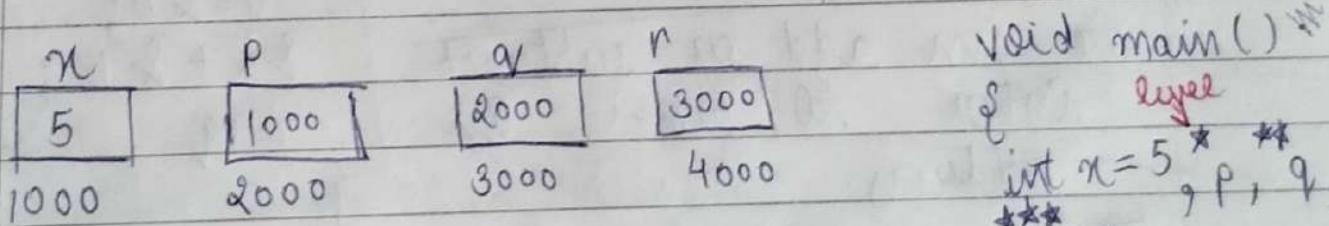
Jab hum * j likhenge to vo a hogा.

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 sirf 2000 pata hai vo float type ka hai isliye usko aage ke bits ke bare me bhi pata chal jaiga.

ENTENDED CONCEPT OF POINTERS



a, q ka
maki
m ka

lik lne kam ka hi save kar
sakte hai jaise p, x (lvl 0)
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

* $q = p$

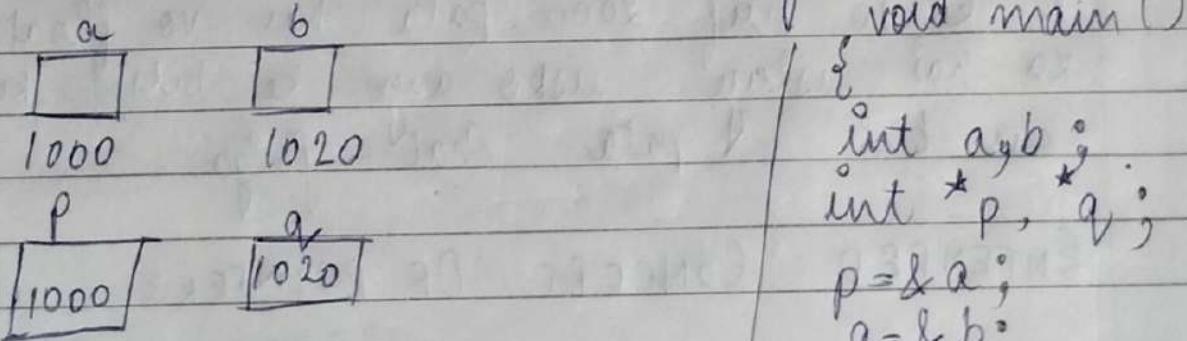
** $q = 7;$ \boxed{x}
 7
 1000

*** $u = 7$

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

$p+1$

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
 $q-p$ $1020 - 1000 = 20 / 2 = 10$ int type
 $p-q$ -10

- Pointer - pointer = Literal subtraction / size of (type of pointer)

APPLICATION OF POINTERS:-

ACTUAL AND FORMAL ARGUMENTS.

function
to declare
kaise kare
kaise kai

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

```
main()
```

```
{
```

```
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
apne x
aur y
me change
karenge
nto a, b me
ni hoga.

4

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);
swap (&a, &b);
printf ("a=%d", a, b);
getch();

void swap ()

{

int t;
t = a;
a = b;
b = t;

}

galat
hai

value \rightarrow
change
kaise kuch
ni hoga.

a ki value aai hai a 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 nahi hoga.

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

void swap (int *x, int *y)

{

int t;
t = *x;
*x = *y;
*y = t;

}

CALL By REFERENCE

PAGE NO. / /
DATE: / /

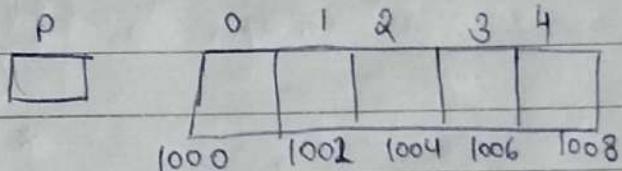
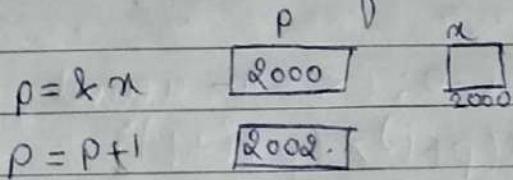
- Call by reference is same as call by address
- When formal arguments are point variables, it is call by reference
- Reference means address

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 ($i=0$; $i \leq 4$; $i++$)

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

for ($i=0$; $i \leq 4$; $i++$) $p + i$

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

pointer se array ko point karte sau blocks ke excess kai.

PAGE NO. _____
DATE: / /

```
main()
{
    void input (int *p)
    {
        int a[5];
        clrscr();
        input (a);
    }

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

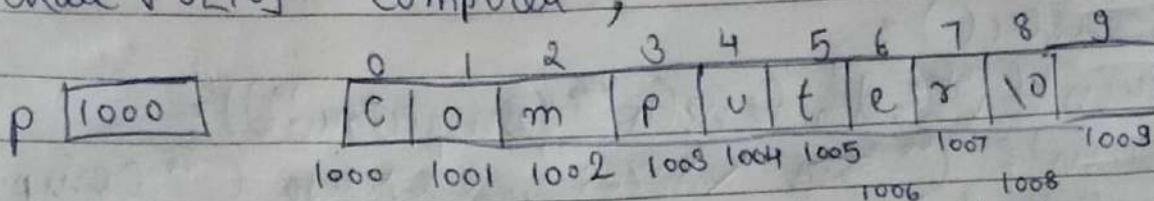
main()

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

POINTERS AND STRING

~~X~~ Chan pointer

- String is stored in char array
 - $\text{char } s[10] = \text{"Computer"};$



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

~~STRING~~ CONSTANT

- String Literal = String Constant = String

Double quotes
Ke and au
wala ko

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

• printf (s);

String length calculate

studen (& s[0]): Address pass kann sehr hei.

on
sheles ("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()

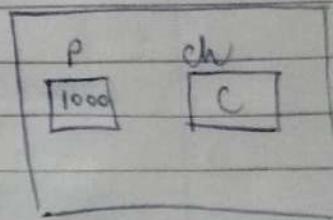
COMPUTER

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

PAGE NO. / /
DATE: / /

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

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



POINTERS AND STRUCTURE IN C LANGUAGE

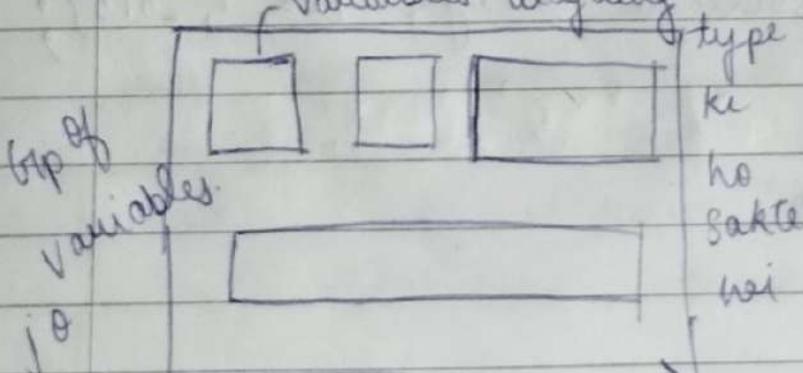
Ques What is structure?

- 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



will ~~can~~ info store kar re hain Non Primitive

Structure
ki madad
se banta
hai
datatype

DEFINING A STRUCTURE

struct tag {
 {
 field1
 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.

Ques Fara 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 likhkar. (def)
lagana padega.

d,

e.g) struct date

{

int d, m, y;

{;

void main ()

{

struct date today; d, ;

today. d = 26 ;

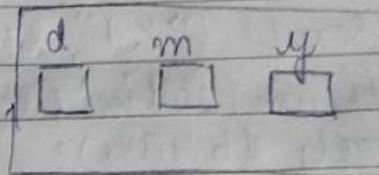
today. m = 7 ;

today. y = 2020 ;

d1. d = today. d ;

d1. *m = today. m ;

d1. y = today. y ;



or

d1 = today ;

Agar user se input
karni 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
chan vagan me sunf ek ek info hi store ho
paig na ki puri isliye hum struct use
kar sakte hai. Bahut sare variables ke liye

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

PAGE NO. _____
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 है

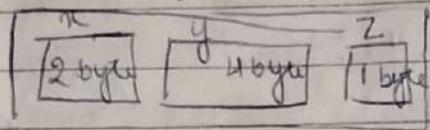
| | |
|----------|------|
| PAGE NO. | icon |
| DATE | / / |

DIFF B/W UNION AND STRUCT

Memory ka diff hota hai mainly

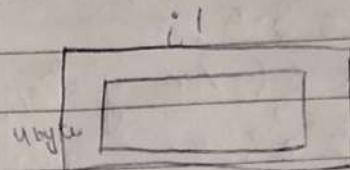
struct item

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



union Item

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



Yaha par je
variable sabse
jaya ho
to hoga
i1 ko utni
memory
me aay
milegi

Paaantu apko teeno variables data ki ek saath
jaaveat hai to aap struct hi banaiye.

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

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

(e.g) union Item

```
int x, float y, char z;
};

void main()
{
```

union Item i;

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

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

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

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

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

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

```
getch();
```

Harne se pass 1 byte
memory mili ho
1 byte ka kis
valuet store kar
na hai

x me kuch raha
vo thi usi block
me

B

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 ej hoti

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 khetam, a destroy ho jaiga. SVA me
ye ni kar sakte ki a phle hi destroy
ho jai.

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

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

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. Aun 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 uski

if 10 nos enter karega then we use SNA.
agar humne nahi pata ki user kitne nos enter
karega 50, 100 ya 5000 tab hum PAGE NO.
Karenge DATE: DNA ka use

DNA

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

malloc() function ko hum call karenge. It's not a declaration statement, its an action statement. Compiler ko samaj ni jaiga malloc(). Run pe ke time sun 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 ligi. malloc me hum variable ka data type ni bata rahe hai, hum sikhna seedha uska size bata rahe hai.

e.g) main()

P
→

3.4

float * p;

if ()

*p = malloc(4)

*p = 3.4

Malloc ka written type kya hai? Void.

malloc je address return karta hai, vo int type

kahya float type ka name nahi pata hai.

PAGE NO. 31 | DATE: / /

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

float ka hi address store hona chahiye. Malloc to void return karnega na. Jo garbhd ho jaigi to apan kya karenge? To hum karenge typecast. Typecast ka matlab hai jo value malloc return kar raha hai, uska nante change kar re hαι uska type jo hai uske change kar re hαι.

p = (float *) malloc(4)

↳ float block ka address hoga jo malloc return karnega.

- Malloc() VS calloc()

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

int *p;



p = calloc(5,2);

Hain block 2-2 byte ~~ka~~ hain

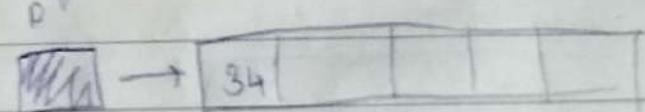
Malloc()

- ① ki maddad se banta hai ek block

calloc()

- 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 barabar ~~blocks~~ agar bad me uske size change karne ki jarurat pade to

realloc()

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

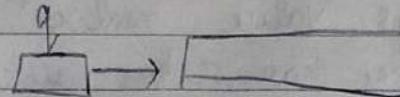
PAGE NO. icon
DATE: / /

double * q;

q = realloc(pter, 8);

float * pte;

pter = malloc(4);



pter
pter → 3.5 * pter = 3.5

free() ka kaam hei memory release karna.

Eg) main()

void fun()

{

int x; // local variable SMA

int * p;

p = malloc(2);

x ka kam hota hi
memory release ho
jaogi block khatam
hoti hai.

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

p kabhi bhi SMA se bane variables ki memory
release ni karta. Vo sifir DMA se bane
variables ki memory release karta hai.

Memory leak ho jai.

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 mili hain

main() {

enum month m1, m2, m3

 { Inne 12 values store kar sakte hain

jo humne phle hi bata di hain.

- 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 seetha m1 = 3 likh dete to bahut confusion ho jaegi 3 & march ko sakta hain ya april bhi ko sakta hain

- 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
enum boolean
{
 false, true
};

PAGE NO. _____
DATE: / /

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

Q Check a number is even or odd?

Ans

```
enum boolean isEven (int n) {  
    if (n % 2 == 0)  
        return (true);  
    else  
        return (false);  
}
```

```
void main ()
```

```
{  
    int n;  
    enum boolean result;  
    clrscr();  
    printf ("Enter a number");  
    scanf ("%d", &n);  
    if (result = isEven (n));  
        if (result == true)  
            printf ("Even number")  
        else  
            printf ("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;
};
```

{
 struct student

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

}
 void main()

}
 struct student s1;

OR

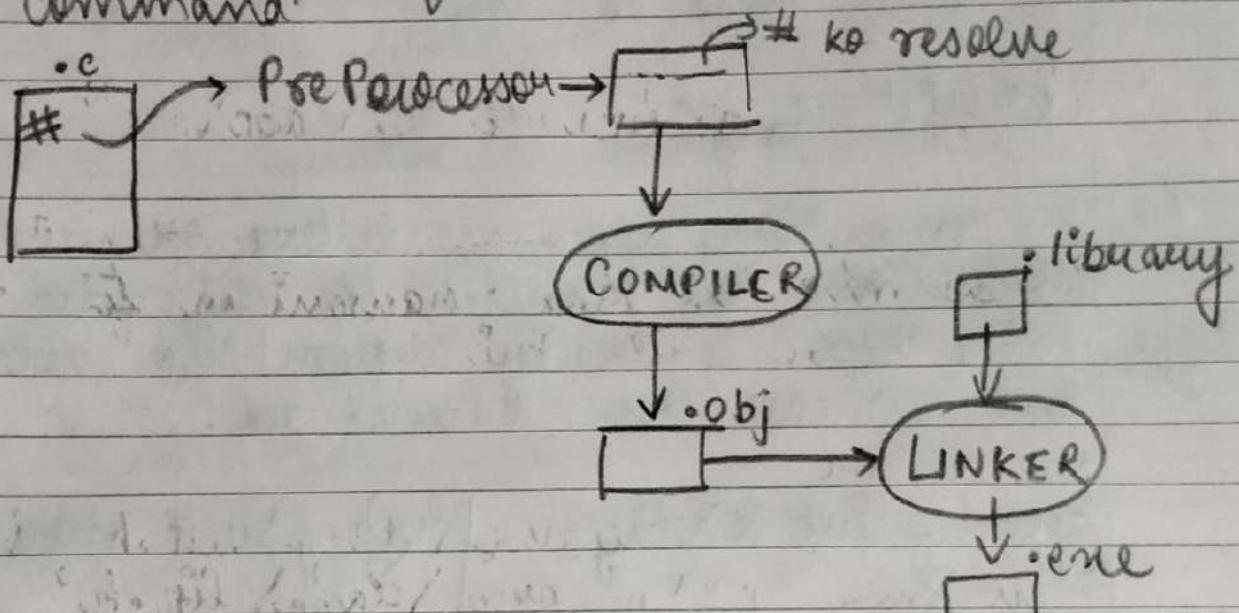
typedef struct Student

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

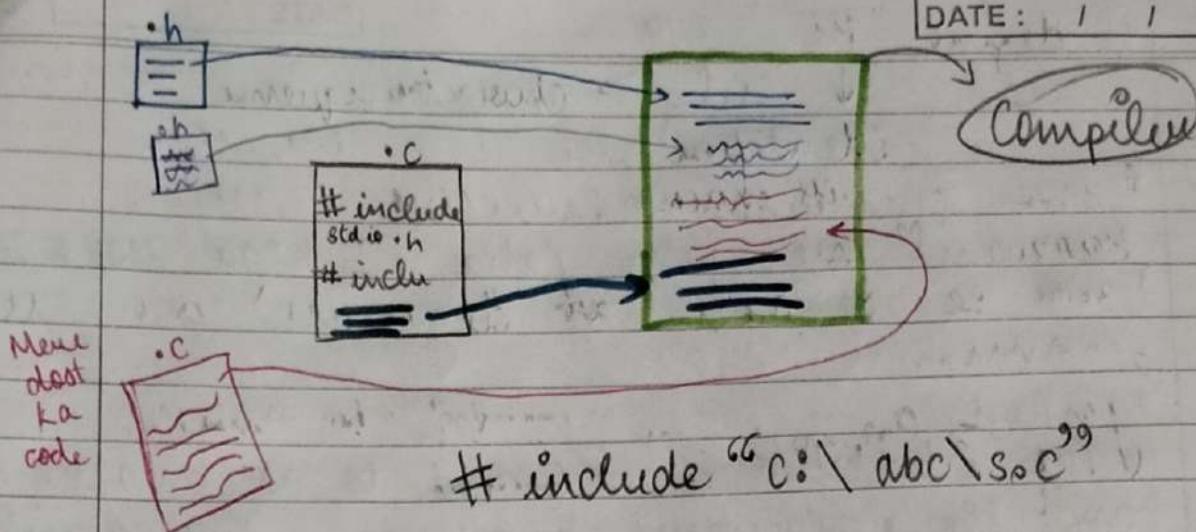
{
 STUDENT s1;
 s1. roll no = 34;
 strcpy(s1. name, "SARUBH");
 s1. age = 20;

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 jayada # 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
jaewali 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 ↓ 3.14 → character sequence
identifier
in 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

e.g.) # 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

e.g.) # 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 icon
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+2, 4-6)

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

Prec Processor Commands

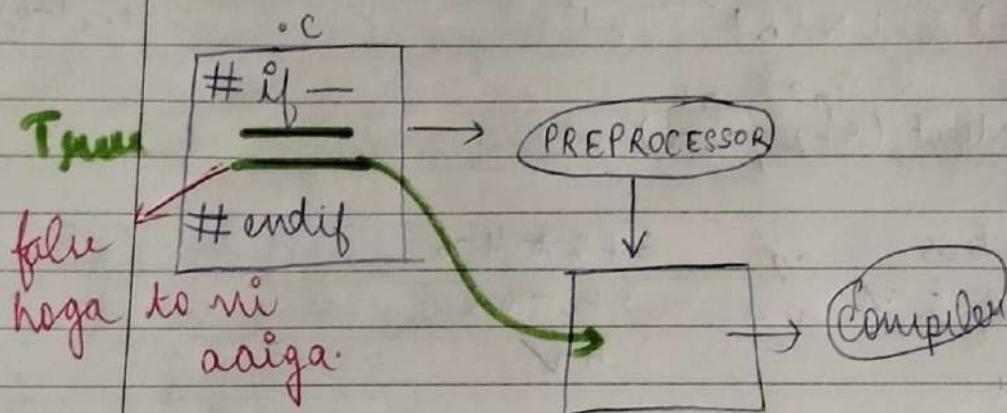
- # if, # else, # elif, # endif
- # ifdef ; # ifndef
- ##

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 macros

- If the macro has been defined by a

PAGE NO. 33
DATE: / /

#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.

e.g) #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 ~~to define~~ 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));

}

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

$$34 + 12 = 46$$