

~~literals~~

constant \rightarrow doesn't change

variable \rightarrow may change

(or) identifier

keyword \rightarrow special meaning,
reserved words.

constants:

Primary

Int

Real

char

Array

String

Pointers

Structures

Uncons.

Rules: (Int) 4 bytes

- Int:
- 1) at least 1 digit
 - 2) no decimal
 - 3) $\times 10^0$, +ve, -ve
(sign).

range -2147483648 to 2147483647

$0 \rightarrow$ octal
 $0x/0X \rightarrow$ Hexa
decimal

Real

(4294967296).

1) same

2) decimal

3) same.

4) exponent - mantissa

0.000342

3.42×10^{-4} / E

+ve/-ve

separated

at least digit

+ve/-ve

-3.4×10^{-38} to 3.4×10^{38}

char

single alphabet, digit, symbol

' , ' , 'A', '5', '='

Vars:-

Results of calc, are stored here (in mem).
to make retrieval & usage clear/easy,
they are named.

Values keep on changing, so called
variables.

Rules:-

comb of alpha, num, underscores.
1st ↓
 not 1st

decl:-

int si;

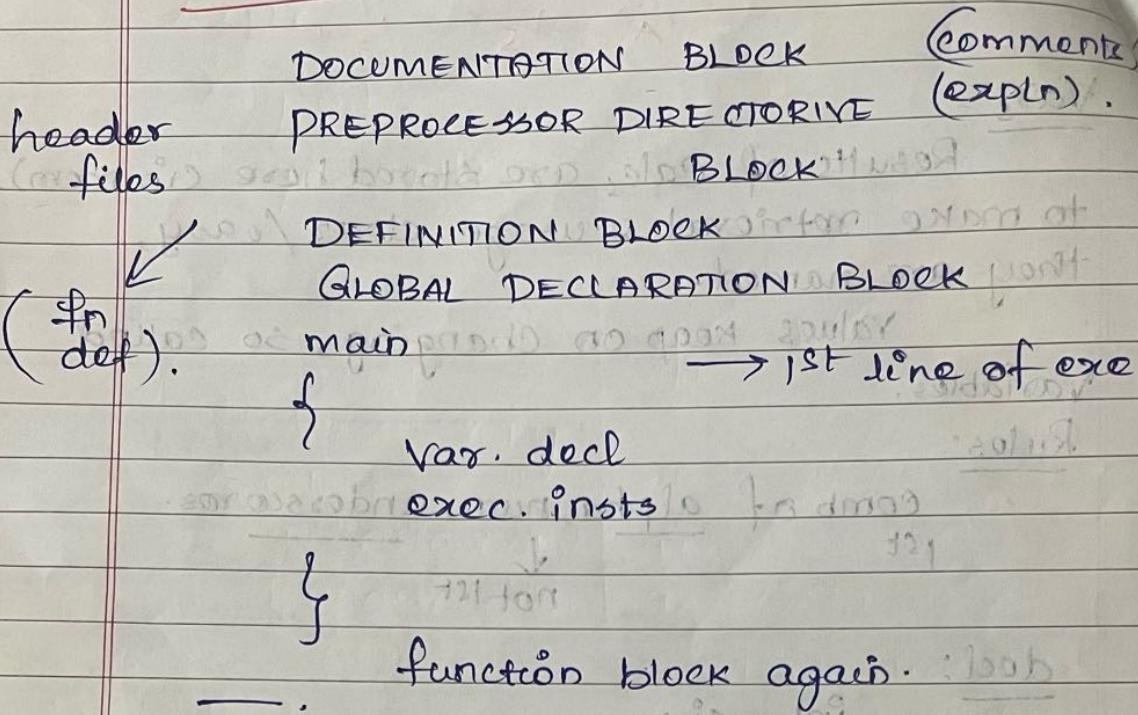
float basic;

char code;

Keywords:- (32)

auto	double	int	struct
break	else	long	switch
case	enum	register	typedef
char	extern	return	union
continue	float	short	unsigned
default	for	signed	void
do	goto	sizeof	volatile
const	if	static	while

Structure of C



Rules:

end with ;

lower cases

Comments:

/* */

optionally placed //

block separator

*/

main()

→ function

container for a set of statements.

int main()

{

return 0; → success.

}

Pointf → can print (3, 3+2, e,
a+b * c-d);

`scanf()` → blank, tab, new line

↓
spacebar ↓ tab ↓ enter key.

②)	'3.15'	x
	35,550	x
	3.2502	✓
	'e Learning'	x
	4652	x
	B'day	x
	int	x
	\$hello	x
	main()	x
	totalArea	✓
	Variable name	x

Instructions:

- 1) Type `decln inst` → to declare types of vars.
- 2) Arith , ,
- 3) control , ,

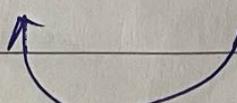
1) `int i=10;` → corr
`int bas;` → decl. $b = 1.99 + 2.4 * 1.44;$
`float b=a+3.1, a=1.5;` X
`int a,b,c,d;`
`a=b=c=d=10;` ✓ `int a=b=c=d=10` X

2) `Var (=) Var | consts`

only one var



Operands.



Value stored on left side