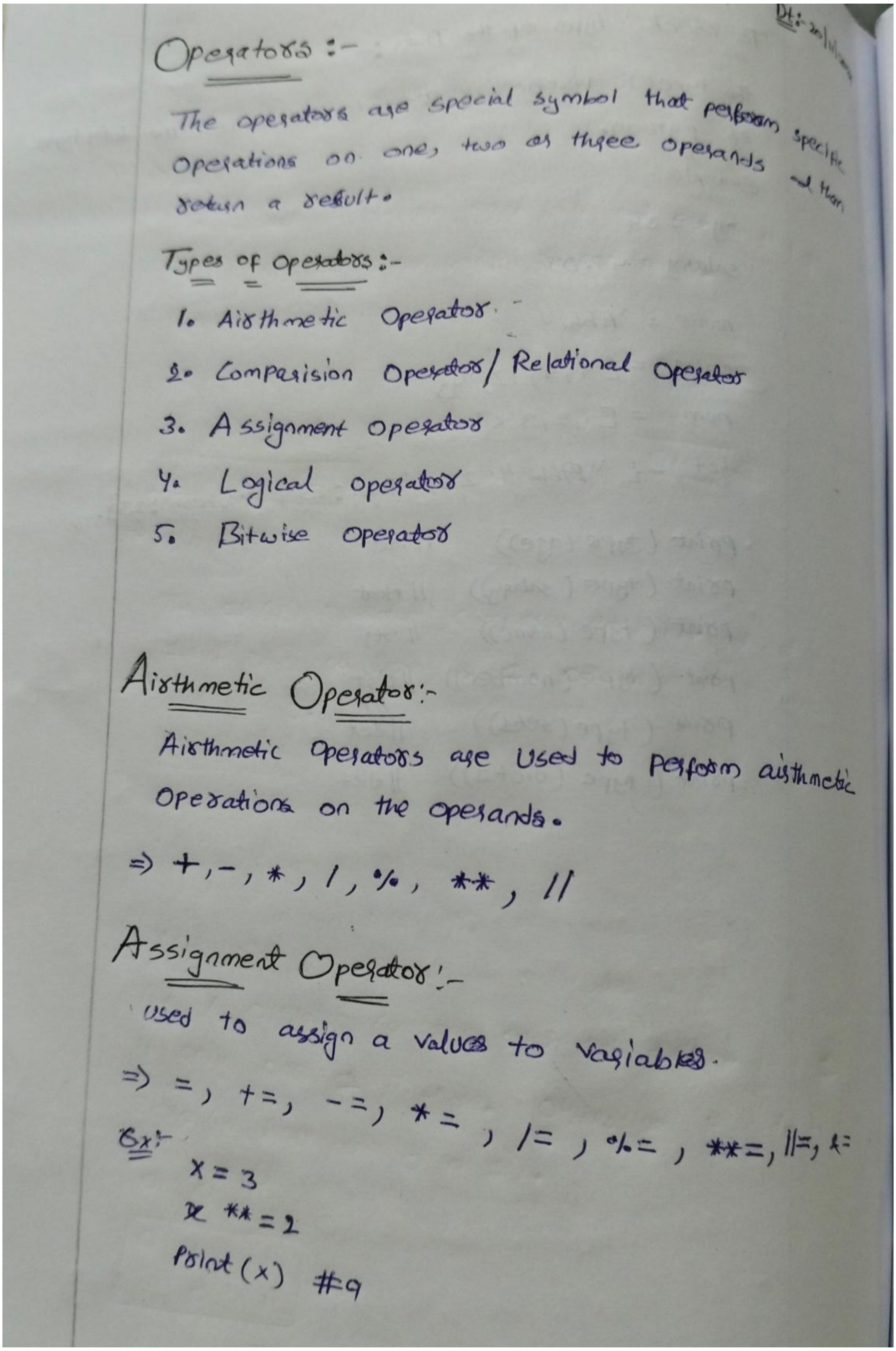
Operators



Dt:- 20/11/

Operators :-

The operators are special symbol that perform specific operations on one, two or three operands and then return a result.

Types of operators :-

1. Airthmetic Operator.
2. Comparision Operator / Relational Operator
3. Assignment Operator
4. Logical operator
5. Bitwise operator

Airthmetic Operator:-

Airthmetic Operators are Used to perform airthmetic Operations on the operands.

⇒ +, −, *, /, %, **, //

Assignment Operator:-

Used to assign a values to variables.

⇒ =, +=, −=, *=, /=, %=, **=, //=, &=

Ex:-

```
x = 3
x **= 2
Print (x)  #9
```

Relational /comparision Operators:-

- Used to compare values. They return a boolean value true or False.
- ==, >, <, !=, >=, <=

Logical Operator:-

Logical operator returns a boolean value by Evaluating boolean Expression.

→ and
→ or
→ not.

Logical And:-

		o/p
⇒ true	true	true
⇒ true	False	False
⇒ False	true	False
⇒ false	false	false

Logical OR:-

		o/p
⇒ true	true	true
⇒ true	false	true
⇒ false	true	true
⇒ false	false	false.

Logical Not:-

	o/p
true	False
False	true

Bitwise Operator:-

Bitwise opeletors ale used to compare (binary) numbers

⇒ AND (&)

The operator compare each bit ad set it to 1 if both are 1 other wise it set to 0.

		o/p
0	0	0
0	1	0
1	0	0
1	1	1

Ex:- 6&3

6 = 110

3 = 011

010 - ②

⇒ OR(|)

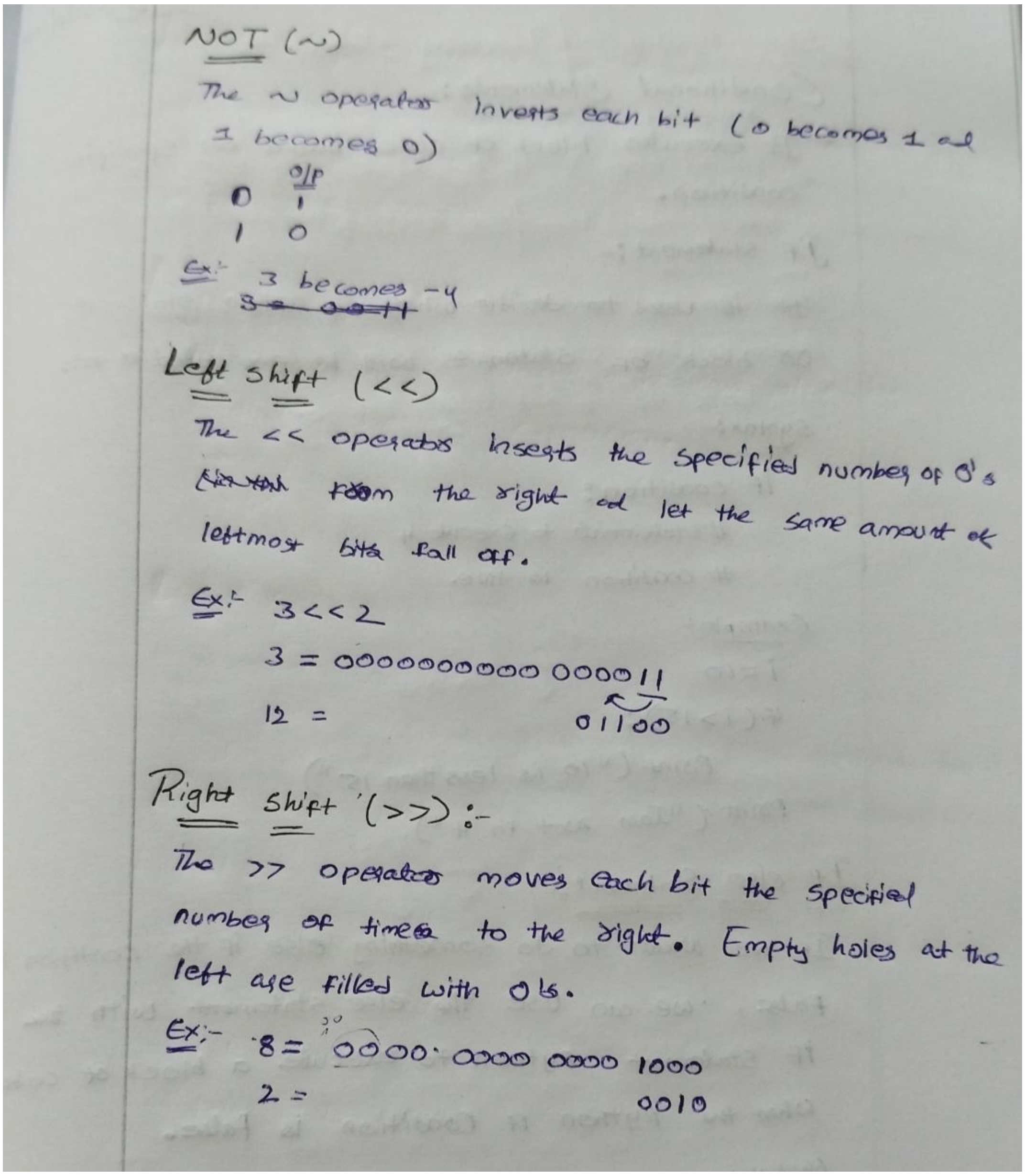
The OR(|) operator compares each bit ad set it to 1 if one or both is 1. Otherwise it returns set to 0.

		o/p
0	0	0
0	1	1
1	0	1
1	1	1

⇒ XOR (^)

The ^ operator compares each bit ad set it to 1 if only one '1' is there. Otherwise it seto to zero (if both are 0 (a) 1).

		o/p
0	0	0
0	1	1
1	0	1
1	1	0



NOT (~)

The ~ operator inverts each bit (0 becomes 1 ad 1 becomes 0)

	O/P
0	1
1	0

Ex:- 3 becomes -4
~~3 = 0011~~

Left shift (<<)

The << operator inserts the specified number of 0's ~~[illegible]~~ from the right ad let the same amount of leftmost bits fall off.

Ex:- 3<<2

3 = 0000000000 00011

12 = 01100

Right shift (>>) :-

The >> operator moves each bit the specified number of times to the right. Empty holes at the left are filled with 0's.

Ex:- 8 = 0000 0000 0000 1000

2 = 0010