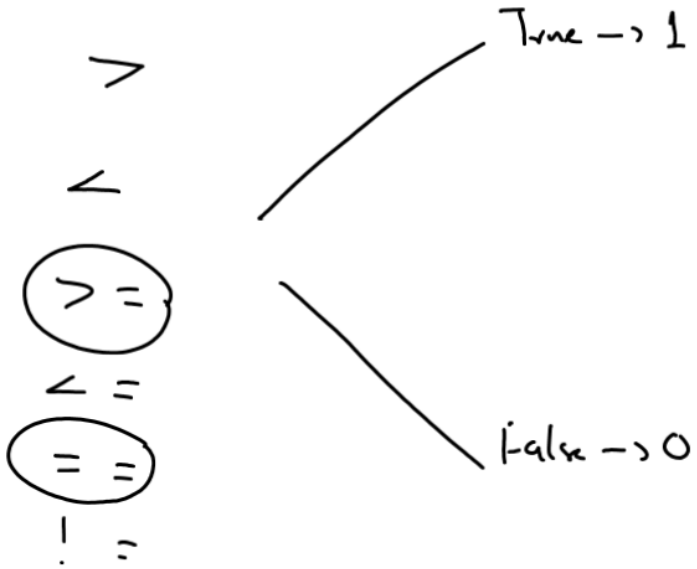
Relational Operators



= Vs ==

① Assignment operator

```
int a=10, b=7;

a = b;

printf("%d %d",a,b);
   7  7
```

① Equality Testing operator

```
int a=10, b=7, c;

c = a == b;
    10 == 7

printf("%d %d %d",a,b,c);
   10  7  0
```

True → 1
False → 0

(j)

```
int a=5, b=9;
int c;
c = a>b;
printf("%d %d %d", a, b, c);
```

5 9 0

```
int a=5, b=9;
int c;
c = a<b;
printf("%d %d %d", a, b, c);
```

5 9 1

Multi - Assignment Exp

```
int a=10, b=7, c;

c = a = b;

printf("%d %d %d", a,b,c);
        7 7 7
```

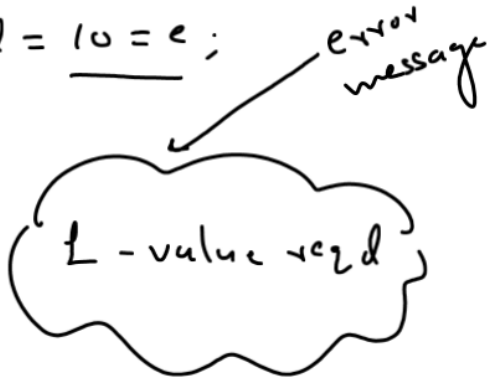
Another example :

```
int a, b, c, d, e;

a = 10;
b = 10;
c = 10;   OR   a=b=c=d=e=10;
d = 10;
e = 10;

printf("%d %d %d %d %d", a,b,c,d,e);
        10 10 10 10 10
```

Point To Remember

```
int a, b, c, d, e;
```

✗ `a = b = c = d = 10 = e;`



✗ `10 = 11;`

✗ `10 = 10;`

✓ `10 == 11;`

0

✓ `10 == 10;`

1

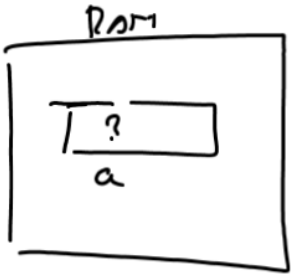
-32768 To 32767

```
int a, b, c, d, e;

a = b = c = d = e;

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

garbage value

```
int a;
close();
printf("%d", a);
```

Any random value can be printed!



```
int a=10, b=10;
int c;
```

① ✓ `c = a = b;`
10 10

② `c = a = 'b';`
98

③ ✗ `c = 'a' = 'b';`

④ ✗ `c = 'a' = b;`

b ⟶ variable

v/s

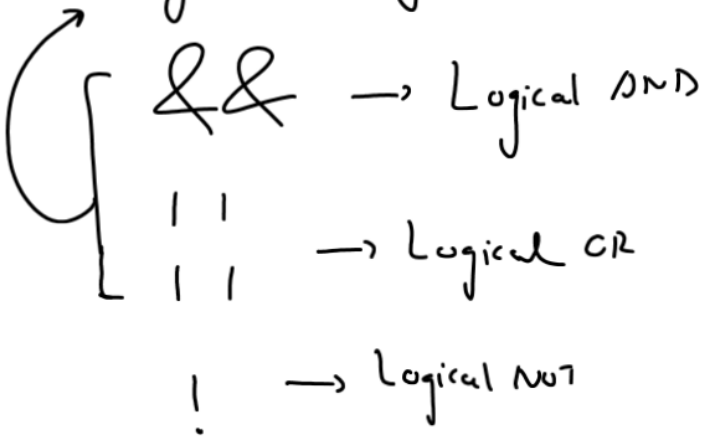
'b' ⟶ char const with ASCII code 98

v/s

b( ) ⟶ might be a fn

Logical Operators

(Used for checking multiple conditions)



X $0 < n < 10$

X $a > b < c$

Cond 1	&&	Cond 2	Result
T	&&	T	T
T	&&	F	F
F	&&	T	F
F	&&	F	F

Cond 1	||	Cond 2	Result
T	||	F	T
F	||	T	T
T	||	T	T
F	||	F	F

```
int a = 10, b = 5, c = 15, d;
```

```
d = a > b && b > c;
    1       0
0
```

```
d = a > b || b > c;
    1       0
1
```

```
int a = 10, b = 10, c = 10, d;
```

Non Sense!

```
d = a == b == c;
    1    == 10
0
```

```
int a = 10, b = 10, c = 10, d;
```