① & → And

2 & 3 == 2

```
  0010
& 0011
------
  0010  ⇒ 2
```

2 & 4 == 0

```
  0010
& 0100
------
  0000  ⇒ 0
```

② | → Bitwise OR

2 | 4 == 6

```
  0010
| 0100
------
  0110  ⇒ 6
```

2 | 3 == 3

```
  010
| 011
-----
  011  ⇒ 3
```

(iii) ~ → Bitwise Not

2 ⇒ 0 1 0

~2 ⇒ ~(0 0 1 0)

1 1 0 1

(iv) XOR ⇒ exclusive or

1 ^ 1 = 0

0 ^ 1 = 1

1 ^ 0 = 1

0 ^ 0 = 1

2 ^ 3 == 1

0 0 1 0

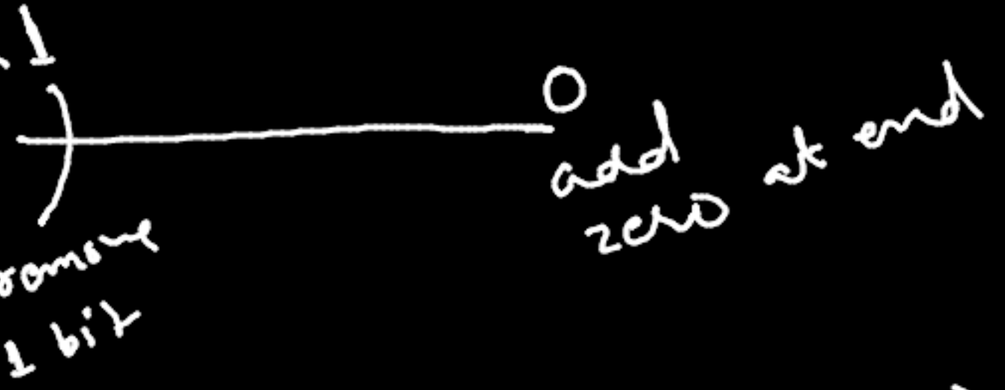
^ 0 0 1 1

0 0 0 1 ⇒ 1

(v) <<

left shift

eg 42<<1



eg. 2 <<1 ⇒4

0 0 1 0 0

0 1 0 0 ⇒ 4

15<<1 ⇒30

0 0 1 1 1 1

0

2^7 2^3 2^2 2^1

0 1 1 1 1 0 ⇒ 30

16+8+4+2

eg. 15 >>1 ⇒ 7

0 1 1 1 1

+ 0

0 0 1 1 1

15 >>2 ⇒ 3

0 0 1 1 1 1

00 +

0 0 0 0 1 1 ⇒ 3

Increment and Decrement

$x = x + 1 \Rightarrow x++$

$y = y - 1 \Rightarrow y--$

$x++$ → Assign the ↑

$++x$ → ↑ then assign

pre increment

eg. $x = 10$

$y = ++x$ | $y = x++$ Post increment

cout<<y → 11

cout<<x → 11

cout<<y → 10

cout<<x → 11

① Break and ② continue

↓

Break the loop & conditions.

↓

skip that part

③ Scope of a variable

→ close are having more scope

and not

$3n+2$, %4 $x = 10$

$n=1$, 5 .

$n=2$ 8 (X)

$n=3$ 11 .

$n=4$ 14.

$n=5$ 17.

$n=6$ 20 (X)

$n=7$ 23 .

$n=8$ 26.

29.

32 (X)

35.

38.

41.

n = 1250 Reverse Number

rev = 521

n = 1250 → 0

rev = 0

while (n != 0)

temp = n % 10;

rev = rev * 10 + temp

n /= 10;

temp	0	5	2	1
rev	0	5	52	521
n	1250	125	12	1	0

Dec To Binary

2	12	0 ✓
2	6	0
2	3	1
	1	

1100

2	10	0
2	5	1
2	2	0
	1	

2	7	1
2	3	1
	1	

⇒ 111

2	8	0
2	4	0
2	2	0
	1	

0101

1010 ✓

1010

int i = 1;

(14)

$i * i \leq n$

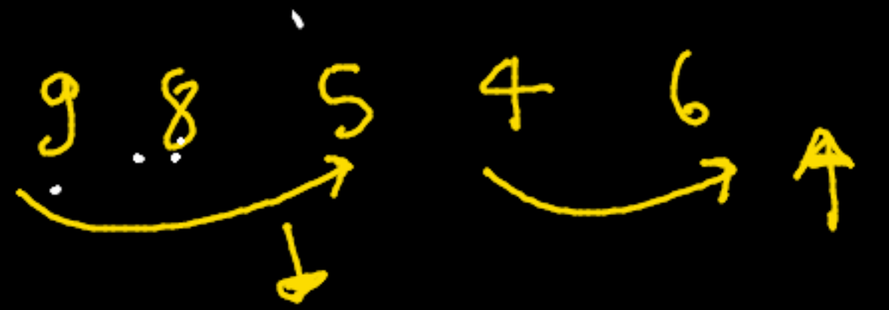
$1 * 1 \leq 14$ → (i) = 2

$2 * 2 \leq 14$ = (i = 3)

$3 * 3 \leq 14$ = (i = 4)

Strictly ↓ then strictly ↑

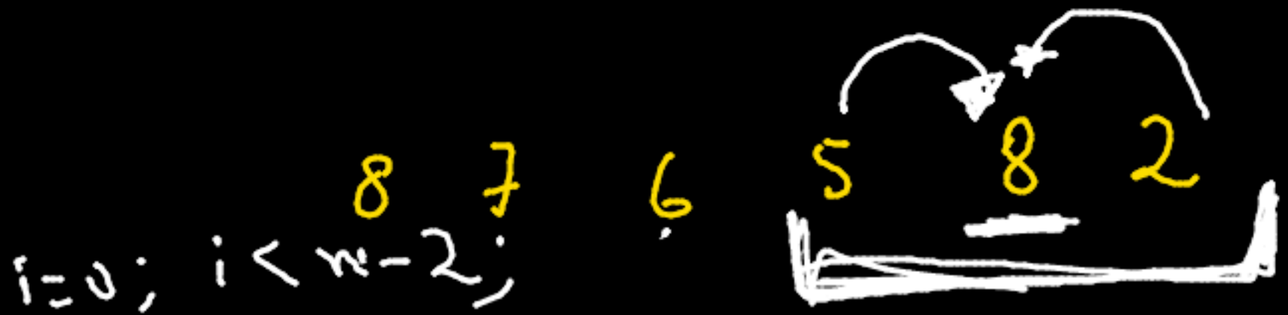
9 8 5 4 6 true



1 2 3 ↑ true

8 7 7 false

8 7 6 5 4 2 ↓ (X) false



false

if (arr[i] ≤ arr[i+1] &&
arr[i+1] ≥ arr[i+2]
return false;

Good Question

* check number sequence
* deci to binary

Point :- to_string(x)
(x is int)