Loops

Entry control

- for
- while

Exit control

- do while

loop condition

```
for(int i=0; i<=10; i++) {
```

initialise

increment/decrement

→ `for(;;)`

```
}
```

```
while (i<=10) {

}
```

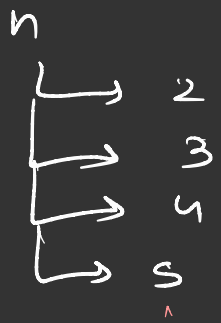
```
do {

} while (i<=10)
```

```
if ( 10 > 20)
    Syso(10)
```

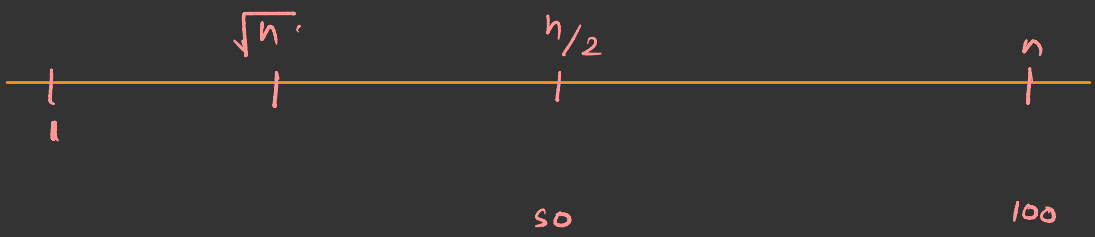
```
for(int i=0; i<10; i++)
        syso(i);        ] - 10 times
        syso(10);       ] → 1 time
```

Ques n is Prime

only gets divided by 1 and itself




```
boolean isPrime = true;
for(int i=2; i*i<=n ; i++) {
    if ( n % i == 0) {
        isPrime = false;
        break;
    }
}
if (isPrime) syso ("Prime");
else syso ("Not prime");
```



$i <= \sqrt{n}$

$i^2 <= (\sqrt{n})^2$

$i * i <= n$

```
for (int i=1; i<5; i++) {
    if (i==3)
        continue;
    syso(i);
}
```

i = 1 2 3 4

```
1
2
4
```

```
if (i==3);
```

Patterns

Ques $n = 4$

```
    1 2 3 4
1   4
2   3 3
3   2 2 2
4   1 1 1 1
```

$$ele + r = n + 1$$

$$ele = n + 1 - r$$

Ques $n = 4$

```
    1 2 3 4
1   4
2   4 3
3   4 3 2
4   4 3 2 1
```

rows = n

col = row_no

$$e + col = n + 1$$

$$e = n + 1 - col$$

Ques n = 4

	1	2	3	4
1	1			
2	2	3		
3	4	5	6	
4	7	8	9	10

```
int count = 1
for(r=1; r<=n; r++) {
    for(c=1; c<=r; c++)
        syso(count++);
    syso();
}
```

Ques n = 4

	1	2	3	4
1	-	-	-	*
2	-	-	*	*
3	-	*	*	*
4	*	*	*	*

$\Rightarrow r + c \leq n$

rows = n

col = 4

'-' = n - # '*'

'*' = row_no

'-' = n - row

```
for(r=1; r<=n ; r++) {
    for(c=1 ; c<=n ; c++) {
```

```
    }
}
```

	1	2	3	4
1	*	-	-	-
2	*	*	-	-
3	*	*	*	-
4	*	*	*	*

$C <= r$ *

Ques 2 2.1 ⇒ 4

$$5.716 \times 100 = \frac{571 \times 2}{100}$$