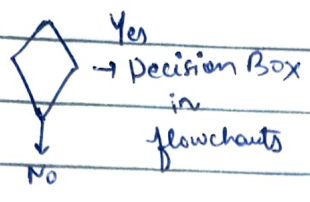


Conditional and Loops



⇔ To take a condition and check that condition

If-else block
 ↓
 if for condition true
 ↘ else otherwise

```
int main() {
    int a, b;
    cout << "Enter two numbers";
    cin >> a >> b;
    if (a == b) {
        cout << "Hey they are equal";
    }
    else {
        if (a < b) {
            cout << "a is smaller";
        }
        else {
            cout << "b is smaller";
        }
    }
}
```

Note point

Variations of If-else

1) Only if block → only for true block

```
if ( ) {
    }
}
```

2)

```
if ( ) { } → output
if ( ) { } → output
else { }

```

 They are separate statements

- * Comments are written in the code to make it more readable & understandable to the user/developer.
- * It is basically not printed on the screen basically written to give an edge to the developer regarding the info about the code what we are doing in it.

* Single line comment → //

* Multiline comment → /* */

if-else if-else

```
if() { }  
else if() { }  
else { }
```

As many else if statements are there that means any no. of else if statements could be written.

```
[ if(a==b) { cout << "They are equal"; }  
  else if(a<b) { cout << "a is smaller"; }  
  else { cout << "a is greater"; }
```

Even-Odd

```
int main() {  
    int num;  
    cin >> num;  
    if(num % 2 == 0) { cout << "Even"; }  
    else { cout << "Odd"; }  
}
```

MCQs

- 1) Hello Hi
- 2) Inside if 15
- 3) $var1 = 5$
 $var2 = 6$
 $cout << 2$
- 4) $a \leq b$ evaluates to true but $!b$ evaluates to false (! operator negates the value, b is non-zero and hence !b is non-zero makes 0 which is false)

Check Case

```

int main() {
    char ch;
    cin >> ch;
    if (ch >= 'A' && ch <= 'Z') {
        cout << "1"; }
    else if (ch >= 'a' && ch <= 'z') {
        cout << "0"; }
    else {
        cout << "-1"; }
}

```

Understanding While loop

```

while ( ) {
}

```

$N \rightarrow 1 \dots N$

int i = 1; \rightarrow initialisation

while (i <= N) { \rightarrow condition

 cout << i;

 i++; \rightarrow update

}

More on while loop

Prime No Code

problem for tick mark

☒ \rightarrow divided

```

int n; cin >> n;
int d = 2; bool divided = false;
while (d <= n) {
    if (n % d == 0) {
        divided = true; }
    d++; }
if (!divided) {
    cout << "Prime"; }
else {
    cout << "Not Prime"; }

```

$N = 7$

$d = 2$ to 6

$\downarrow +1$

3 to 6

$\downarrow +1$ to 6

4 to 6

$\downarrow +1$

5 to 6

$\downarrow +1$

6 to 6

WOW!

Print Sum of Even Numbers from 1 to N

```
int n; cin >> n;
int sum = 0, i = 1;
while (i <= n) {
    if (i % 2 == 0) {
        sum += i;
    }
    i++;
}
cout << sum << endl;
```

Start, End, Step $\rightarrow ((f-32) * 5) / 9 = C$

Patterns

→ 1
→ 1 2
→ 1 2 3
→ 1 2 3 4

ith row has i columns

```
i = 1
while (i <= n) {
    j = 1
    while (j <= i) {
        cout << " ";
        j++;
    }
    cout << endl;
    i++;
}
```

```
int n;
cin >> n;
int i = 1;
while (i <= n) {
    int j = 1;
    while (j <= i) {
        cout << j << " ";
        j++;
    }
    cout << endl;
    i++;
}
```

N = 3

3 <= 3

j = 1 2 <= 1 j = 1 2

1 <= 1 3 <= 2

j = 1

4 <= 3

1	2		
1	2	3	
1	2	3	4

→	1
→	2 3
→	4 5 6
→	7 8 9 10

N	i	j	val(k)
3	$\frac{n+1}{2}$ 3	$\frac{n+1}{2}$ $\frac{n+1}{2}$	$\frac{n+1}{2}$ 4 8

```

int n;
cin >> n;
int i = 1; int k = 1;
while (i <= n) {
    int j = 1;
    while (j <= i) {
        cout << k; k++;
        j++;
    }
    cout << endl;
    i++;
}

```

↓
2 3 2
4 5 6

#

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

→ spaces + tent
code code

```

int n; cin >> n;
int i = 1; int val = 1;
while (i <= n) {
    int j = 1;
    while (j <= n - i + 1) {
        cout << " ";
        j++;
    }
    int k = 1;
    while (k <= i) {
        cout << val;
        val++;
        k++;
    }
    cout << endl;
    i++;
}

```

WOW!!!

Return Statement is a special keyword when encountered ends the main. That means, no statement after it will be executed when it is encountered.

Power of Number

• $25 \rightarrow n$

↓
x {

int count = 1; int ans = 1;

while (count <= n) {

ans * = x;

count ++;

}

cout << ans;

}

END OF TOPIC