FREEMIND

Date

Page

Lecture-6 (Loops)

→ Types of Loops
- While loop
- For loop
- Do-while loop

While loop

```
while (condition) {
    // code          (code runs till condition is true)
}
```

For Loop

```
for (initialization; condition; final statement) {
                                  (increment)
    //code
}
```

⇒ Omitting parts of for loop

1) Omitting init.-statement

```
int i = 0          → (initialize before loop)
for ( ; i<5 ; i++ ) {
    cout << i;
}
```

2) Omitting condition.

```
for ( i=1 ;      ; i++ ) {
    //code   → {should include any stopping condition}
}                (otherwise there is an infinite loop)
```

3) Omitting final expression

```
for (int i=1 ; i<5 ;  ) {
    cout << i;
    i++;          ——→ (final expression in code)
}
```

→ When we have to use more than one var. in a loop

```
for (int i=0, j=4; i<4, j>0; i++, j--) {

}
```

(A) for loop (1) Vs while loop (2)

⇒
```
for (initialization; condition; increment) {
    //code
}
```
(1)

⇒
```
initialization
while (Condition) {
    //code
    final expression
}
```
(2)

(1) While loop ko use karne ka benefit tab hai jab hame condition k bare me pata na ho. Tab hame loop to input k according stop karvana ho.

FREEMIND

Date

Page

eg:

```
while (true) {
    int n;
    Cin >> n;
    If (n == -1)
        //stop loop (break statement)
    else
        cout << n;
}
```

break keyword :-
Take us out from the loop/~~Nearest loop~~

Q1 -> Print the 1st multiple of 5 which is also a multiple of 7.

Sol ->

```
int i = 5;
while (true) {
    if (i % 7 = 0) {
        cout << i << endl;
        break;
    }
    i += 5
}
return 0;
}
```

```
for (int i = 5; ; i += 5) {
    if (i % 7 == 0) {
        cout << i << endl;
        break;
    }
}
```

(★) Do-while loop ⇝ code will work atleast once whatever the condition is true/fals

```
do {
    //code
} while (condition);
```

Q2 Print the sum of ~~the~~ the stream of N integers in the input using do-while loop.

Sol ->

```
int n;
Cin >> n;
int sum = 0
```

★

```
do {
  int num
  cin>> num;
  Sum += num;
  n--;
} while (n>0);

cout << sum << endl;

return 0;
}
```

⭐ Continue keyword → skip to the iteration.

{Works to the nearest loop}.

```
int i=1
  while (i<6) {
    if (i==3) {
      continue;
    cout << i;
    }
```

2] Output

```
int i=0
for (int i=1; i<5; i++) {
}
cout << i;
```

output = 0

```
int a = 1
while (a<4) {
  cout << a << " ";
  a++;
}
```

output: 1 2 3

Date
Page

Lecture-7 Problems on loops - Part-1

Q⇒ Pattern Printing: Rectangular Pattern

① * * * * * *
* * * * * *
* * * * * *

rows, columns
3 , 6
↑ ↑
n m

```
for (int i=1; i<=3; i++) {
    for (int j=1; j<=6; j++) {
        cout << "*";
    }
    cout << endl;
}
```

rows(i)	columns(j)	Print?
1 to n	1 to m	*

② (Hollow Rectangle)

Column 1 Column m
Row 1 → * * * * * *
* *
* *
Row n → * * * * * *

```
int main() {
    int n, m;
    cin >> n >> m;
    for (int i=1; i<=n; i++) {
        for (int j=1; j<=m; j++) {
            if (i==1 || j==1 || i==n || j==m) {
                cout << "*";
            }
            else {
                cout << " ";
            }
        } cout << endl;
    }
    return 0
}
```

③

(i) rows	(j) Columns
1 to n	1 to Row no.

Print "*"

```
int main() {
    int n;
    cin >> n;
    for (int i=1; i<=n; i++) {
        for (int j=1; j<=i; j++) {
            cout << "*";
        }
        cout << endl;
    }
    return 0;
}
```

{Triangular Pattern}

④ Inverse ~~Triangular~~ right Triangular ▽

```
row 1  * * * *
    2  * * *
    3  * *
    4  *
```

	rows (i)	columns (j)
Urvi mam	1 to n (i++)	1 to (n − row no. + 1)
Aayush	n to 1 (i−−)	1 to row no. (i) (j++)

```
int main () {
    int n;
    cin>>n;
    for (int i=n; i>0; i--) {
        for (int j=1; j<=i; j++) {
            cout<<"*";
        }
        cout<<endl;
    }
    return 0;
}
```

⑤

row	pattern	Space	Stars		rows
1	_ _ _ *	3	1	2(1) − 1	1 to n
2	* * *	2	3	2(2) − 1	
3	_ * * * * *	1	5	2(3) − 1	
4	* * * * * * *	0	7	2(4) − 1	

Columns: " " → 1 to (n − row no.); "*" → 1 to (2i − 1)

```
int n;
cin>>n;
for (int i=1; i<=n; i++) {
    for (int j=1; j=(n-i); j++) {
        cout<<" ";
    }
    for( int j=1; j<=(2*i - 1); j++) {
        cout<<"*";
    }
    cout<< endl;
}
```

(6) Numerical Rectangular Pattern

```
1 2 3 4 5 6 7
2 3 4 5 6 7 1
3 4 5 6 7 1 2
4 5 6 7 1 2 3
5 6 7 1 2 3 4
6 7 1 2 3 4 5
7 1 2 3 4 5 6
```

rows: 1 to n

Columns: [i to n] + [1 to (i-1)]

```
int n;
Cin>>n;
for(int i=1; i<=n; i++){
    for(int j=i; j<=n; j++){
        Cout<<j;
    }
    for(int j=1; j<=(i-1); j++){
        Cout<<j;
    }
    Cout<<endl;
}
```

(7)

```
1 2 3 4 5 6
1 2 3 4 5 6
1 2 3 4 5 6
```

rows: 1 to n

Colums: 1 to m

```
int n,m;
Cin>>n>>m;
for(int i=1; i<=n; i++){
    for(int j=1; j<=m; j++){
        Cout<<j;
    }
    Cout<<endl;
}
```

Date
Page

(8)

```
1 2 3 4 5 6
1       6
1       6
1 2 3 4 5 6
```

rows: 1 to n
columns: 1 to m

```
int n, m;
cin>>n>>m;
for(int i=1; i<=n; i++){
    for(int j=1; j<=m; j++){
        if(j==1 || i==1 || j==m || i==n){
            cout<<j;
        }
        else{
            cout<<" ";
        }
    cout<<endl;
    }
}
```

(9)

```
1 2 1 2 1 2
2 1 2 1 2 1
1 2 1 2 1 2
2 1 2 1 2 1
```

(i+j)=2 (even) ⟶ 1
(i+j)=3 (odd) ⟶ 2

n ↓ 4 (rows) m ↓ 6 (columns)

```
int n, m;
cin>>n>>m;
for(int i=1; i<=n; i++){
    for(int j=1; j<=m; j++){
        if((i+j)%2==0){
            cout<<"1";
        }
        else{
            cout<<"2";
        }
    }
    Count<<endl;
}
```

(10)
```
1
1 2
1 2 3
1 2 3 4
```

Rows: 1 to n

Columns: 1 to row no.

(11)
```
    1
  1 2 1
1 2 3 2 1
1 2 3 4 3 2 1
```

Spare	No.
3	
2	
1	
0	

Rows: 1 to n

Columns: 1 to i | 1 to (i-1)

(i)

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

(12)
```
    1
  2   2
3       3
4 4 4 4 4 4 4
```

Lecture-8 {Problems on Loops - Part-2}

Q⇒ Count the no. of digits for a given number n.

```
int main(){
    int n;
    cin>>n; //610
    int digits=0;
    while(n>0){
        digits++;
        n=n/10;
    }
    cout<<digits<<endl;
    return 0;
}
```

610/10 = 61 , 1
61/10 = 6 . 2
6/10 = 0 (3)
3 digits

Q⇒ Find the sum of digits in a given number n.

Sol:⇒
```
int n;
cin>>n;
int sum=0;
while(n>0){
    int lastdigit = n%10;
    sum+= lastdigit;
    n=n/10;
}
cout<<sum<<endl;
```

Q⇒ Find the reverse of a no.

```
int n;
cin>>n;
int reverse=0;
while(n>0){
    int lastdigit = n%10
    reverse = reverse*10 + lastdigit
    n/=10
```

FREEMIND

Date

Page

Q⇒ Find sum of the following series $S = 1-2+3-4 \ldots n$

Sol:-
```
int n;
cin>>n;
int result = 0;
for (int i = 0; i<=n; i++) {
    if (i%2 == 0) {
        result -= i
    } else {
        result += i;
    }
}
cout << result << endl;
```

```
1 to n
if (num == odd)
        sum += num
else
        sum -= num
```

Q⇒ Print the first n factorial numbers

Sol:-
```
int n;
cin>>n;
int fact = 1;
for (int i = 1; i<=n; i++) {
    fact = fact * i;
    cout << fact << endl;
}
return 0;
```

Q⇒ Given 2 no.s a and b. Find a raise to the power b. (a^b)

```
int a, b;
cin >> a >> b; int result = 1
for (int i = 1; i<=b; i++) {
    result = result * a
}
cout << result << endl;
```