PATTERNS

Date : / /
Page No.

① Solid rectangle

```
  1 2 3 4 5
1 * * * * *
2 * * * * *
3 * * * * *
```

User input → no. of rows
↓
no. of lines

no. of columns
↳ no. of things in each line

Q) To print 5 stars in n rows

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

⇒ Now take m as no. of stars in each line

Eg. n=4, m=3 →
```
* * *
* * *
* * *
* * *
```

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

Date : / /
Page No.

⇒ **Solid Square**

user input → n

```
* * * * ..... n
* * * * ..... "
* * * * ..... "
* * * * ..... "
| | | |
| | | |
n n n n
```

→
```
int main (){
    int n;
    cout << "Enter no. of rows & columns";
    cin >> n;
    for(int i=1; i<=n; i++){
        for(int j=1; j<=n; j++){
            cout << "*" ;
        }
        cout << endl;
    }
}
```

→ ~~Triangle~~ Q→ 1 2 3 4
1 2 3 4
(Number 1 2 3 4 ⇒ n=4
Square) 1 2 3 4

Date : / /
Page No.

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

⇒ Triangle Patterns

① Star triangle

```
1  *
2  * *        n=4
3  * * *
4  * * * *
```

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

Date : / /
Page No.

② Star Triangle reverse

```
* * * *
* * *
* *
*
```

n=4

```
1  2  3  4
```

→1 * * * * → 4 stars

2 * * * → 3 "

3 * * → 2 "

4 * → 1 "

⇒ Row no. + no. of stars = n+1

no. of stars = n+1-i

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

Date : / /
Page No.

③ Number Triangle

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

→
```
for (int i=1; i<=n; i++){
    for (int j=1; j<=i ; j++){
        cout << j ;
    }
    cout<<endl ;
}
```

→ To print →
```
1 2 3 4
1 2 3
1 2
1
```

Put $j <= n+1-i$ instead $j <= i$

④ Odd Number Triangle

```
1
1 3          { n=4 }
1 3 5
1 3 5 7
```

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

Date : / /
Page No.

OR

```
for(int i=1; i<=n; i++){
    int a=1;
    for(int j=1; j<=i; j++){
        cout<<a;
        a+=2;
    }
    cout<<endl;
}
```

⇒ Alphabet Pattern

① Alphabet Square

```
A B C D
A B C D
A B C D
A B C D
```

⇒ n=4

ASCII of A = 65
" " B = 66
" " C = 67

```
for(int i=1; i<=n; i++){
    for(int j=1; j<=n; j++){
        cout<<(char)(j+64)<<" ";
    }
    cout<<endl;
}
```

Date : / /
Page No.

⇒ **Star Plus**

```
    1 2 3 4 5
1   _ _ * _ _
2   _ _ * _ _
3   * * * * *      n=5
4   _ _ * _ _
5   _ _ * _ _
```

⇒ In this pattern we are only printing * when row and column no. is equal to the mid of n.

```
int main(){
    int n;
    cin >> n;
    for(int i=1; i<=n; i++){
        for(int j=1; j<=n; j++){
            if(i == n/2+1 || j == n/2+1) cout<<"*";
            else cout<<" ";
        } cout<<endl;
    }
}
```

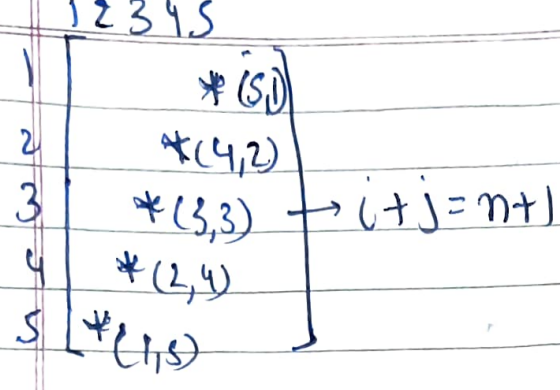
→ **Star Cross**

```
  1 2 3 4 5
1 *     *
2   *   *      n=5
3     *
4   *   *
5 *     *
```

```
    1   2   3   4   5
1   *(1,1)
2       *(2,2)
3           *(3,3)
4               *(4,4)
5                   *
                  (5,5)
```

Row no. = Column no.

Date : / /
Page No.




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

⇒ Floyd's Triangle

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

extra variable
↳ declare a variable outside of the loop
→ then increment it by 1 after every ~~iterate~~ print.

→
```
int main(){
    int n, k;
    cin >> n;
    k=1;
    for(int i=1; i<=n; i++){
        for(int j=1; j<=i; j++){
            cout<<k;
            k++;
        }
    }
```

Date : / /
Page No.

⇒ Binary Triangle

```
  1 2 3 4
1 1
2 0 1
3 1 0 1      n=4
4 0 1 0 1
```

⇒ Method-1 (Using Extra Variable)

if (i%2==0) → 1 se start
else → 0 se start

& alternate 0 & 1

→
```
int a;
for(int i=1; i<=n; i++){
    if (i%2==0) a=0; // row no. even
    else a=1; // row no. odd
    for(int j=1; j<=i; j++){
        cout << a << " ";
        // flipping
        if (a==1) a=0; } or a=1-a
        else a=1;
    }
}
```

⇒ Method-2

if (i==j) → 1
if i & j both odd → 1
if i & j both even → 1
else → 0

Date : / /
Page No.

⇒ Star Triangle Flipped

```
1 2 3 4
1 _ _ _ *
2 _ _ * *
3 _ * * *
4 * * * *
```

spaces and stars
↓ ↓
loop loop

```
_ _ _ *
_ _ * *
_ * * *
* * * *
```

→
```
for (int i=1; i<=n; i++){
    //spaces
    for (int j=1; j<=n-i; j++){
        cout<<" ";
    }
    for(int k=1; k<=i; k++){
        cout<<"*";
    }
    cout<<endl;
}
```

⇒ Rhombus

```
_ _ _ * * * *   → 3 space, 4 star
_ _ * * * *     → 2 "   , 4 star
_ * * * *       ⇒ n=4 → 1 space, 4 star
* * * *                 → 0 space, 4 star
```

```
_ _ _ *
_ _ * *
_ * * *
* * * *
```

Date : / /
Page No.

```
for (int i=1; i<=n; i++){
    //Spaces
    for (int j=1; j<=n-i; j++){
        cout<<" ";
    }
    for (int k=1; k<=n; k++){
        cout<<"*";
    }
}
```

⇒ Star Pyramid

```
   *
  ***
 *****
*******
```

n=4 ⇒

```
   *
  ***
 *****
*******
```

Odd star triangle

→ Odd star triangle →

```
1  *          → 1
2  * * *      → 3        ⇒ j=1 to 2i-1
3  * * * * *  → 5
4  * * * * * * * → 7
```

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

Date : / /
Page No.

(*) For Pyramid

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

⇒ Number Pyramid Palindrome

1	1	1	1	
2	1 2 1	2	1 2 +	1
3	1 2 3 2 1	3	1 2 3	2 1
4	1 2 3 4 3 2 1	4	1 2 3 4	3 2 1

→ To print - ① 1
② 2 1 ⇒ (j = i; j >= 1; j--)
③ 3 2 1 cout << j;

⇒ Number pyramid Palindrome

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

Date : / /
Page No.

⇒ Star Pyramid Diamond

```
1        *          ⇒ i = 1 to i <= 2*n-1
2       ***
3      *****        n = 4
4     *******
5      *****
6       ***
7        *
```

nsp = no. of spaces = n-1 → 3 → nsp-- / nsp++
nst = no. of stars = 1 → nst+=2 / nst-=2
↳ till 4th line ↳ After 4th line

→
```
int nst=1
int nsp=n-1
for(int i=1; i<=2*n-1; i++){
    //Spaces
    for(int j=1; j<=nsp; j++){
        cout<<" ";
    }
    if(i<=n-1) nsp--;
    else nsp++;
    //Stars
    for(int k=1; k<=nst; k++){
        cout<<"*";
    }
    if(i<=n-1) nst+=2;
    else nst-=2;
    cout<<endl;
}
```

Date : / /
Page No.

⇒ **Number spiral**

```
4 4 4 4 4 4 4
4 3 3 3 3 3 4
4 3 2 2 2 3 4
4 3 2 1 2 3 4   ⇒ n = 4
4 3 2 2 2 3 4
4 3 3 3 3 3 4
4 4 4 4 4 4 4
```

⇒ Inverse of the above spiral

(1) ←

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

each i,j cell has a value min(i,j)

→ For i = 1 1 1 1
(1) 1 2 2 2 →
1 2 3 3
1 2 3 4

```
for(int i=1; i<=n; i++){
    for(int j=1; j<=n; j++){
        cout<< min(i,j);
    }
    cout endl;
}
```

→ This will not work if either the row or column exceeds 4.
i.e for 5, 6 & 7.

Date : / /
Page No.

We can bypass it if somehow we convert the 5th row no. 5, 6 & 7 to 3, 2 & 1 respectively by using pseudo row no.

And same for the column no.

→ We can apply if condⁿ if (a > n) (pseudo row no.)

a = 2 * n - i ;