

L5 : Pattern 2 Practice Questions

1-Tut : Code : Mirror Number Pattern

[Send Feedback](#)

Print the following pattern for the given N number of rows.

Pattern for N = 4

```
...1
..12
.123
1234
```

The dots represent spaces.

Input format : Integer N (Total no. of rows)

Output format : Pattern in N lines

Constraints 0 <= N <= 50

Sample Input 1:3

Sample Output 1:

```
1
12
123
```

Sample Input 2:4

Sample Output 2:

```
1
12
123
```

```
1234
```

```
1. #include<iostream>
2. using namespace std;
3.
4.
5. int main(){
6.
7.     /* Read input as specified in the question.
8.        * Print output as specified in the question.
9.        */
10. int N;
11. cin >> N;
12. int i = 1;
13. while(i <= N){
14.
15.     int spaces = 1;
```

```

16.     while(spaces <= N-i){
17.         cout << " ";
18.         spaces++;
19.     }
20.
21.     int n =1;
22.     while(n <= i){
23.         cout << n;
24.         n ++;
25.     }
26.     cout << endl;
27.     i++;
28. }
29. }

```

2-Tut : Code : Inverted Number Pattern

[Send Feedback](#)

Print the following pattern for the given N number of rows.

Pattern for N = 4

```

4444
333
22
1

```

Input format : Integer N (Total no. of rows)

Output format : Pattern in N lines

Constraints : $0 \leq N \leq 50$

Sample Input 1:5

Sample Output 1:

```

55555
4444
333
22
1

```

Sample Input 2:6

Sample Output 2:

```

666666
55555
4444
333
22
1

```

1. `#include<iostream>`
2. `using namespace std;`
3. `int main(){`
- 4.
5. `/* Read input as specified in the question.`

```

6.          * Print output as specified in the question.
7.          */
8.  int N ;
9.  cin >> N;
10. int i =1;
11. while(i <= N){
12.     int p = N-i+1;
13.     int k =1;
14.
15.     while(k <= p){
16.         cout << p;
17.         k++;
18.     }
19.     cout << endl;
20.     i++;
21. }
22. }

```

3-Tut : Code : Star Pattern

[Send Feedback](#)

Print the following pattern

Pattern for N = 4

```

. . . ★
. . ★★
. ★★
★★★★
★★★★

```

The dots represent spaces.

Input Format : N (Total no. of rows)

Output Format : Pattern in N lines

Constraints : 0 <= N <= 50

Sample Input 1 : 3

Sample Output 1 :

```

*
***
*****

```

Sample Input 2 : 4

Sample Output 2 :

```

*
***
*****
*****

```

```

1. #include<iostream>
2. using namespace std;
3. int main(){
4.
5.     /* Read input as specified in the question.
6.        * Print output as specified in the question.
7.        */
8.     int N ;
9.     cin >> N;
10.    int i =1;
11.    while(i <= N){
12.        int spaces = 1;
13.        while(spaces <= N-i){
14.            cout << " ";
15.            spaces ++ ;
16.        }
17.        int star = 1;
18.        while(star <= 2*i-1){
19.            cout << "***";
20.            star ++;
21.        }
22.        cout << endl;
23.        i++;
24.    }
25. }

```

4-Tut : Code : Triangle of Numbers

[Send Feedback](#)

Print the following pattern for the given number of rows.

Pattern for N = 4

```

• • • 1
• • 232
• 34543
4567654

```

The dots represent spaces.

Input format : Integer N (Total no. of rows)

Output format : Pattern in N lines

Constraints : $0 \leq N \leq 50$

Sample Input 1:5

Sample Output 1:

```

1
232
34543
4567654
567898765

```

Sample Input 2:

Sample Output 2:

1

232

34543

4567654

```
1. #include <iostream>
2. using namespace std;
3.
4. int main() {
5.     /* Read input as specified in the question.
6.      * Print output as specified in the question.
7.      */
8.     int N;
9.     cin >> N;
10.    int i =1;
11.    while(i <= N){
12.        int space= 1;
13.        while(space <= N-i){
14.            cout << " ";
15.            space ++;
16.        }
17.
18.        int k =i;
19.        int j = 1;
20.        while(j <= i){
21.            cout << k;
22.            j++;
23.            k++;
24.        }
25.
26.        j =1;
27.        k = 2*(i-1);
28.        while(j <= i-1){
29.            cout << k;
30.            k--;
31.            j++;
32.        }
33.        cout << endl;
34.        i++;
35.    }
36. }
37.
```

5-Tut : Code : Diamond of stars

[Send Feedback](#)

Print the following pattern for the given number of rows.

Note: N is always odd.

Pattern for N = 5

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

The dots represent spaces.

Input format :

N (Total no. of rows and can only be odd)

Output format :

Pattern in N lines

Constraints :

1 <= N <= 49

Sample Input 1:

5

Sample Output 1:

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

Sample Input 2:

3

Sample Output 2:

```
*  
 ***  
  *
```

```

1. #include<iostream>
2. using namespace std;
3. int main() {
4.
5.     /* Read input as specified in the question.
6.      * Print output as specified in the question.
7.      */
8.     int N;
9.     cin >> N;
10.    int i = 1;
11.
12.    int n1 = N / 2 + 1;
13.    while(i <= n1){
14.        int space = 1;
15.        while(space <= n1-i){
16.            cout << " ";
17.            space ++;
18.        }
19.        int star = 1;
20.        while(star <= 2*i -1){
21.            cout << "***";
22.            star ++ ;
23.        }
24.        cout << endl;
25.        i++;
26.    }
27.
28.    i = N/2;
29.    while(i >= 1){
30.
31.        int space = 1;
32.        while(space <= n1-i){
33.            cout << " ";
34.            space ++;
35.        }
36.
37.        int star = 1;
38.        while(star <= 2*i -1){
39.            cout << "***";
40.            star ++ ;
41.        }
42.        cout << endl;
43.        i--;
44.    }
45. }

```