



Q1. Print

```

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

```

for  $n = 4$

Answer :

```

1 #include <iostream>
2 using namespace std;
3
4 int main() {
5     int n;
6     cin >> n;
7     int i = 1; // Try printing in reverse (solved in the video)
8     while(i <= n) {
9         int j = 1;
10        while(j <= n) {
11            cout << j << " ";
12            j = j + 1;
13        }
14        cout << endl;
15        i = i + 1;
16    }
17    return 0;
18 }

```

PROBLEMS OUTPUT TERMINAL DEBUG CONSOLE

```

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

```

```

1 #include <iostream>
2 using namespace std;
3
4 int main() {
5     int n;
6     cin >> n;
7     int i = 1;
8     // Try printing in reverse (solved in the video)
9     while(i <= n) {
10        int j = 1;
11        while(j <= n) {
12            cout << j << " ";
13            j = j + 1;
14        }
15        cout << endl;
16        i = i + 1;
17    }
18    return 0;
19 }

```

PROBLEMS OUTPUT TERMINAL DEBUG CONSOLE

```

rk\Coding\Patterns\" ; if ($?) { g++ Pattern.cpp -o Pattern
3
1 2 3
1 2 3
1 2 3

```

Homework: For  $n = 4$ ,

```

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

```

Q2. Print :

```

1 2 3
4 5 6
7 8 9

```

for  $n = 3$

Logic - maintain a variable, incrementing it by 1 after every cout,

Answer :

```

1 #include <iostream>
2 using namespace std;
3
4 int main() {
5     int n;
6     cin >> n;
7     int i = 1;
8     int toPrint = 1;
9     while(i <= n) {
10        int j = 1;
11        while(j <= n) {
12            cout << toPrint << " ";
13            toPrint = toPrint + 1;
14            j = j + 1;
15        }
16        cout << endl;
17        i = i + 1;
18    }
19    return 0;
20 }

```

PROBLEMS OUTPUT TERMINAL DEBUG CONSOLE

```

\Work\Coding\Patterns\" ; if ($?) { g++ P
3
1 2 3
4 5 6
7 8 9

```

```

1 #include <iostream>
2 using namespace std;
3
4 int main() {
5     int n;
6     cin >> n;
7     int i = 1;
8     int toPrint = 1;
9     while(i <= n) {
10        int j = 1;
11        while(j <= n) {
12            cout << toPrint << " ";
13            toPrint = toPrint + 1;
14            j = j + 1;
15        }
16        cout << endl;
17        i = i + 1;
18    }
19    return 0;
20 }

```

PROBLEMS OUTPUT TERMINAL DEBUG CONSOLE

```

rk\Coding\Patterns\" ; if ($?) { g++ Pa
4
1 2 3 4
5 6 7 8
9 10 11 12
13 14 15 16

```

Try printing :

```

9 8 7
6 5 4
3 2 1

```

for  $n = 3$ .

3 2 1

16 15 14 13  
12 11 10 9  
8 7 6 5  
4 3 2 1

Hint: Starting point has  
a relation with n

Q3. Print :

★  
★ ★  
★ ★ ★  
★ ★ ★ ★

for n = 4.

Hint: We are printing '★', row  
number of times where row=1,2,...

Answer :

```
1 #include <iostream>
2 using namespace std;
3
4 int main() {
5     int n;
6     cin >> n;
7     int row = 1;
8     while(row <= n) {
9         int column = 1;
10        while(column <= row) {
11            cout << "★ ";
12            column = column + 1;
13        }
14        cout << endl;
15        row = row + 1;
16    }
17    return 0;
18 }
```

```
1 #include <iostream>
2 using namespace std;
3
4 int main() {
5     int n;
6     cin >> n;
7     int row = 1;
8     while(row <= n) {
9         int column = 1;
10        while(column <= row) {
11            cout << "★ ";
12            column = column + 1;
13        }
14        cout << endl;
15        row = row + 1;
16    }
17    return 0;
18 }
```

Q4. Print :

1  
2 2  
3 3 3  
4 4 4 4

-for n = 4.

(Easy)

Answer :

```
1 #include <iostream>
2 using namespace std;
3
4 int main() {
5     int n;
6     cin >> n;
7     int row = 1;
8     while(row <= n) {
9         int column = 1;
10        while(column <= row) {
11            cout << row << " ";
12            column = column + 1;
13        }
14        cout << endl;
15        row = row + 1;
16    }
17    return 0;
18 }
```

```
1 #include <iostream>
2 using namespace std;
3
4 int main() {
5     int n;
6     cin >> n;
7     int row = 1;
8     while(row <= n) {
9         int column = 1;
10        while(column <= row) {
11            cout << row << " ";
12            column = column + 1;
13        }
14        cout << endl;
15        row = row + 1;
16    }
17    return 0;
18 }
```

Homework - Print :

1  
2 3  
4 5 6  
7 8 9 10

for n = 4.

Answer :

```
1 #include <iostream>
```

7 8 9 10

Answer :

```
1 #include <iostream>
2 using namespace std;
3
4 int main() {
5     int n;
6     cin >> n;
7     int row = 1, toPrint = 1;
8     while(row <= n) {
9         int column = 1;
10        while(column <= row) {
11            cout << toPrint << " ";
12            toPrint = toPrint + 1;
13            column = column + 1;
14        }
15        cout << endl;
16        row = row + 1;
17    }
18    return 0;
19 }
```

PROBLEMS OUTPUT TERMINAL DEBUG CONSOLE

4  
1  
2 3  
4 5 6  
7 8 9 10

Q5 Print 1 for n = 4.  
2 3  
3 4 5  
4 5 6 7

Logic - We are starting from i for every row i.  
Keep incrementing i times for each row.

Answer :

```
1 #include <iostream>
2 using namespace std;
3
4 int main() {
5     int n;
6     cin >> n;
7     int row = 1, toPrint = 1;
8     while(row <= n) {
9         int column = 1;
10        toPrint = row; // start printing from row
11        while(column <= row) {
12            cout << toPrint << " ";
13            toPrint = toPrint + 1;
14            column = column + 1;
15        }
16        cout << endl;
17        row = row + 1;
18    }
19    return 0;
20 }
```

PROBLEMS OUTPUT TERMINAL DEBUG CONSOLE

4  
1  
2 3  
3 4 5  
4 5 6 7

Homework: Solve the above question without using the extra variable toPrint.

Logic - Column starts from 1 for every row. Change it to start from row and go till row + row = row \* 2 (exclusive)

Answer :

```
1 #include <iostream>
2 using namespace std;
3
4 int main() {
5     int n;
6     cin >> n;
7     int row = 1;
8     while(row <= n) {
9         int column = row;
10        while(column <= row * 2) {
11            cout << column << " ";
12            column = column + 1;
13        }
14        cout << endl;
15        row = row + 1;
16    }
17    return 0;
18 }
```

PROBLEMS OUTPUT TERMINAL DEBUG CONSOLE

```
rk\Coding\Patterns\" ; if ($?) { g++ Pattern.cpp
4
1
2 3
3 4 5
4 5 6 7
```

Q6. Print

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

Hint: starting from row number and decrementing.

Logic : column = 1 2 3 4  
row = 1 1  
row = 2 2 1  
row = 3 3 2 1  
row = 4 4 3 2 1

Subtract column number from row number and add 1.  
(row - column + 1)

Answer :

```
1 #include <iostream>
2 using namespace std;
3
4 int main() {
5     int n;
6     cin >> n;
7     int row = 1;
8     while(row <= n) {
9         int column = 1;
10        while(column <= row) {
11            cout << (row - column + 1) << " ";
12            column = column + 1;
13        }
14        cout << endl;
15        row = row + 1;
16    }
17    return 0;
18 }
```

PROBLEMS OUTPUT TERMINAL DEBUG CONSOLE

```
rk\Coding\Patterns\" ; if ($?) { g++ Pattern.cpp -o Pat
5
1
2 1
3 2 1
4 3 2 1
5 4 3 2 1
```



Q7 Print :     A A A     for n = 3.  
              B B B  
              C C C

Answer :

```
4 int main() {
5     int n;
6     cin >> n;
7     int i = 1;
8     while(i <= n) {
9         int j = 1;
10        while(j <= n) {
11            // you can also print 'A' + i - 1 because with an int
12            // 'A' will get typecasted to 65
13            cout << (char)(65 + i - 1) << " ";
14            j = j + 1;
15        }
16        i = i + 1;
17        cout << endl;
18    }
19    return 0;
20 }
```

PROBLEMS OUTPUT TERMINAL DEBUG CONSOLE

```
rk\Coding\Patterns\" ; if ($?) { g++ Pattern.cpp -o Pattern } ; if ($?) {
4
A A A
B B B
C C C
D D D
```

Q8. Print :     A B C     for n = 3  
              A B C  
              A B C     (Homework Code)

Answer :

```
3
4 int main() {
5     int n;
6     cin >> n;
7     int i = 1;
8     while(i <= n) {
9         int j = 1;
10        char ch = 'A';
11        while(j <= n) {
12            cout << ch << " ";
13            ch = ch + 1;
14            j = j + 1;
15        }
16        i = i + 1;
17        cout << endl;
18    }
19    return 0;
20 }
```

PROBLEMS OUTPUT TERMINAL DEBUG CONSOLE

```
rk\Coding\Patterns\" ; if ($?) { g++ Pattern.cpp -o
3
A B C
A B C
A B C
```

Q9 Print :     A B C     for n = 3  
              D E F  
              G H I

Logic is same as with numbers

Answer :

```
4 int main() {
5     int n;
6     cin >> n;
7     int i = 1;
8     char ch = 'A';
9     while(i <= n) {
10        int j = 1;
11        while(j <= n) {
12            cout << ch << " ";
13            ch = ch + 1;
14            j = j + 1;
15        }
16        i = i + 1;
17        cout << endl;
18    }
19    return 0;
20 }
```

PROBLEMS OUTPUT TERMINAL DEBUG CONSOLE

```
rk\Coding\Patterns\" ; if ($?) { g++ Pattern.cpp -o Pa
3
A B C
D E F
G H I
```

Q10 Print :     A B C  
                 B C D  
                 C D E

Logic is same as with numbers     { 'A' + row - 1  
   for row = 1, 2, ...  
   increment this

Answer :

```
4 int main() {
5     int n;
6     cin >> n;
7     int i = 1;
8     while(i <= n) {
9         char ch = 'A' + i - 1;
10        int j = 1;
11        while(j <= n) {
12            cout << ch << " ";
13            ch = ch + 1;
14            j = j + 1;
15        }
16        i = i + 1;
17        cout << endl;
18    }
19    return 0;
20 }
```

PROBLEMS OUTPUT TERMINAL DEBUG CONSOLE

```
4
A B C D
B C D E
C D E F
D E F G
```

```
4 int main() {
5     int n;
6     cin >> n;
7     int i = 1;
8     while(i <= n) {
9         int j = 1;
10        while(j <= n) {
11            char ch = 'A' + i + j - 2;
12            cout << ch << " ";
13            j = j + 1;
14        }
15        i = i + 1;
16        cout << endl;
17    }
18    return 0;
19 }
```

PROBLEMS OUTPUT TERMINAL DEBUG CONSOLE

```
rk\Coding\Patterns\" ; if ($?) { g++ Pattern.cpp -o Pattern
4
A B C D
B C D E
C D E F
D E F G
```

Q11. Print :     A                     for n = 3  
                 B B  
                 C C C

Answer :

```
4 int main() {
5     int n;
6     cin >> n;
7     int i = 1;
8     while(i <= n) {
9         int j = 1;
10        char ch = 'A' + i - 1;
11        while(j <= i) {
12            cout << ch << " ";
13            j = j + 1;
14        }
15        i = i + 1;
16        cout << endl;
17    }
18    return 0;
19 }
```

PROBLEMS OUTPUT TERMINAL DEBUG CONSOLE

rk\Coding\Patterns\" ; if (\$?) { g++ Pattern.cpp -o

4

A

B B

C C C

D D D D

Q12 Print :      A      (Try yourself)

                 B C

                 D E F

                 G H I J

Logic same as with numbers homework question.

Answer :

```
4 int main() {
5     int n;
6     cin >> n;
7     int i = 1;
8     char ch = 'A';
9     while(i <= n) {
10        int j = 1;
11        while(j <= i) {
12            cout << ch << " ";
13            ch = ch + 1;
14            j = j + 1;
15        }
16        i = i + 1;
17        cout << endl;
18    }
19    return 0;
20 }
```

PROBLEMS OUTPUT TERMINAL DEBUG CONSOLE

4

A

B C

D E F

G H I J

Q13 Print :      A      for n=4

                 B C

                 C D E

                 D E F G

Logic is same as numbers.

Answer:

```
4 int main() {
5     int n;
6     cin >> n;
7     int i = 1;
8     while(i <= n) {
9         char ch = 'A' + i - 1;
10        int j = 1;
11        while(j <= i) {
12            cout << ch << " ";
13            ch = ch + 1;
14            j = j + 1;
15        }
16        i = i + 1;
17        cout << endl;
18    }
19    return 0;
20 }
```

PROBLEMS OUTPUT TERMINAL DEBUG CONSOLE

rk\Coding\Patterns\" ; if (\$?) { g++ Pattern.cpp

4

The Babbar Answer

```
1 #include<iostream>
2 using namespace std;
3
4 int main() {
5
6     int n;
7     cin>>n;
8
9     int row = 1;
10
11    while(row <= n) {
12
13        int col = 1;
14
15        while(col <= row) {
16            char ch = ('A' + row + col - 2);
17            cout<<ch;
18            col = col + 1;
19        }
20        cout<<endl;
21        row = row + 1;
22
23    }
```

```

20 }

PROBLEMS OUTPUT TERMINAL DEBUG CONSOLE
rk\Coding\Patterns\" ; if ($?) { g++ Pattern.cpp
4
A
B C
C D E
D E F G

```

```

18         col = col + 1;
19     }
20     cout<<endl;
21     row = row + 1;
22 }
23
24
25 }

```

(Homework)

Q 14 Print :

```

D
C D
B C D
A B C D

```

Logic - We are starting from  $ch = ('A' + rows - 1)$  in row 1.  
 $ch = ('A' + rows - 2)$  in row 2.  
 $ch = ('A' + rows - 3)$  in row 3.  
:  
:  
:  
 $ch = ('A' + rows - i)$  for row  $i$ .

Answer :

```

4 int main() {
5     int n;
6     cin >> n;
7     int i = 1;
8     while(i <= n) {
9         char ch = 'A' + n - i;
10        int j = 1;
11        while(j <= i) {
12            cout << ch << " ";
13            ch = ch + 1;
14            j = j + 1;
15        }
16        i = i + 1;
17        cout << endl;
18    }
19    return 0;
20 }

PROBLEMS OUTPUT TERMINAL DEBUG CONSOLE
rk\Coding\Patterns\" ; if ($?) { g++ Pattern.cpp
4
D
C D
B C D
A B C D

```

Q 15 Print :

```

      ☆
     ☆☆
    ☆☆☆
   ☆☆☆☆

```

for  $n = 4$  (Notice the number of spaces and its relation with the row number,  $i$ )

Logic - for row  $i$ , we are first printing  $n - i$  spaces and then printing  $i$  stars.



Answer :

```
4 int main() {
5     int n;
6     cin >> n;
7     int i = 1;
8     while(i <= n) {
9         int space = 1;
10        while(space <= n - i) {
11            cout << " ";
12            space = space + 1;
13        }
14        int j = 1;
15        while(j <= i) {
16            cout << "*";
17            j = j + 1;
18        }
19        i = i + 1;
20        cout << endl;
21    }
22    return 0;
23 }
```

PROBLEMS OUTPUT TERMINAL DEBUG CONSOLE

\Work\Coding\Patterns\" ; if (\$?) { g++

4

\*

\*\*

\*\*\*

\*\*\*\*

Q16. Print :

★ ★ ★ ★  
★ ★ ★  
★ ★  
★

(Homework)

Answer :

```
1 #include <iostream>
2 using namespace std;
3
4 int main() {
5     int n;
6     cin >> n;
7     int i = 1;
8     while(i <= n) {
9         int j = 1;
10        while(j <= n - i + 1) {
11            cout << " ";
12            j = j + 1;
13        }
14        i = i + 1;
15        cout << endl;
16    }
17    return 0;
18 }
```

PROBLEMS OUTPUT TERMINAL DEBUG CONSOLE

\Work\Coding\Patterns\" ; if (\$?) { g++

4

\* \* \* \*

\* \* \*

\* \*

\*

Q17. Print :

★ ★ ★ ★  
★ ★ ★  
★ ★  
★

for n = 4

(Homework)

Logic : Printing  $i-1$  spaces &  $n-i+1$  stars for  $i^{\text{th}}$  row.

Answer :

```
4 int main() {
5     int n;
6     cin >> n;
7     int i = 1;
8     while(i <= n) {
9         int j = 1, space = 1;
10        while(space < i) {
11            cout << " ";
12            space = space + 1; // space++ is also used
13        }
14        while(j <= n - i + 1) {
15            cout << "*";
16            j = j + 1;
17        }
18        i = i + 1;
19        cout << endl;
20    }
21    return 0;
22 }
```

PROBLEMS OUTPUT TERMINAL DEBUG CONSOLE

\Work\Coding\Patterns\" ; if (\$?) { g++ Pattern.cpp -o Pattern

4

\*\*\*\*

\*\*\*

\*\*

\*

Q18 Print:      1 1 1 1      for n = 4      (Homework)  
                  2 2 2  
                  3 3  
                  4

Answer :

```

4 int main() {
5     int n;
6     cin >> n;
7     int i = 1;
8     while(i <= n) {
9         int j = 1, space = 1;
10        while(space < i) {
11            cout << " ";
12            space = space + 1; // space++ is also used
13        }
14        while(j <= n - i + 1) {
15            cout << i;
16            j = j + 1;
17        }
18        i = i + 1;
19        cout << endl;
20    }
21    return 0;
22 }

```

PROBLEMS OUTPUT TERMINAL DEBUG CONSOLE

```

\Work\Coding\Patterns\" ; if ($?) { g++ Pattern.cpp -o Pat
4
1111
222
33
4

```

Q19 Print:      1      for n = 4      (Homework)  
                  2 2  
                  3 3 3  
                  4 4 4 4

Answer :

```

4 int main() {
5     int n;
6     cin >> n;
7     int i = 1;
8     while(i <= n) {
9         int j = 1, space = 1;
10        while(space <= n - i) {
11            cout << " ";
12            space = space + 1; // space++ is also used
13        }
14        while(j <= i) {
15            cout << i;
16            j = j + 1;
17        }
18        i = i + 1;
19        cout << endl;
20    }
21    return 0;
22 }

```

PROBLEMS OUTPUT TERMINAL DEBUG CONSOLE

```

\Work\Coding\Patterns\" ; if ($?) { g++ Pattern.cpp -o Pat
4
1
22
333
4444

```

Q20. Print:      1      for n = 4  
                  2 3  
                  4 5 6  
                  7 8 9 10

Logic - Print space (n-i) times and print num. Then increment num.

Answer -

```
4 int main() {
5     int n;
6     cin >> n;
7     int i = 1, num = 1;
8     while(i <= n) {
9         int j = 1, space = 1;
10        while(space <= n - i) {
11            cout << " "; // Notice two spaces
12            space = space + 1;
13        }
14        while(j <= i) {
15            cout << num << " ";
16            num = num + 1;
17            j = j + 1;
18        }
19        i = i + 1;
20        cout << endl;
21    }
22    return 0;
23 }
```

PROBLEMS OUTPUT TERMINAL DEBUG CONSOLE

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

Q 21 Print :  
1  
1 2 1  
1 2 3 2 1  
1 2 3 4 3 2 1

for  $n = 4$

Logic - Use 2 while loops for each row  $i$ .  
One for printing  $(n-i)$  " " (double spaces)  
One for printing from 1 to  $i$ .  
Other for printing from  $(i-1)$  to 1.

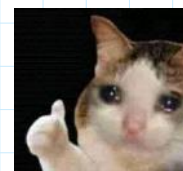
Answer :

```
1 #include <iostream>
2 using namespace std;
3
4 int main() {
5     int n;
6     cin >> n;
7     int i = 1;
8     while(i <= n) {
9         int j = 1, space = 1;
10        while(space <= n - i) {
11            cout << " "; // Notice two spaces
12            space = space + 1;
13        }
14        while(j <= i) {
15            cout << j << " ";
16            j = j + 1;
17        }
18        j = i - 1;
19        while(j >= 1) {
20            cout << j << " ";
21            j = j - 1;
22        }
23        i = i + 1;
24        cout << endl;
25    }
26    return 0;
27 }
```

TAGDA HOMEWORK :

Print :  
1 2 3 4 5 5 4 3 2 1  
1 2 3 4 ★ ★ 4 3 2 1  
1 2 3 ★ ★ ★ ★ 3 2 1  
1 2 ★ ★ ★ ★ ★ ★ 2 1  
1 ★ ★ ★ ★ ★ ★ ★ ★ 1

for  $n = 5$



Answer :

```
4  int main() {
5      int n;
6      cin >> n;
7      int i = 1;
8      while(i <= n) {
9          // part 1: numbers from 1 to n - i + 1
10         int j = 1;
11         while(j <= n - i + 1) {
12             cout << j << " ";
13             j = j + 1;
14         }
15         // part 2: Stars (i-1)*2 times
16         j = 1;
17         while(j <= (i-1)*2) {
18             cout << "x ";
19             j = j + 1;
20         }
21         // part 3: numbers from n - i + 1 to 1
22         j = n - i + 1;
23         while(j >= 1) {
24             cout << j << " ";
25             j = j - 1;
26         }
27         i = i + 1;
28         cout << endl;
29     }
30     return 0;
31 }
```