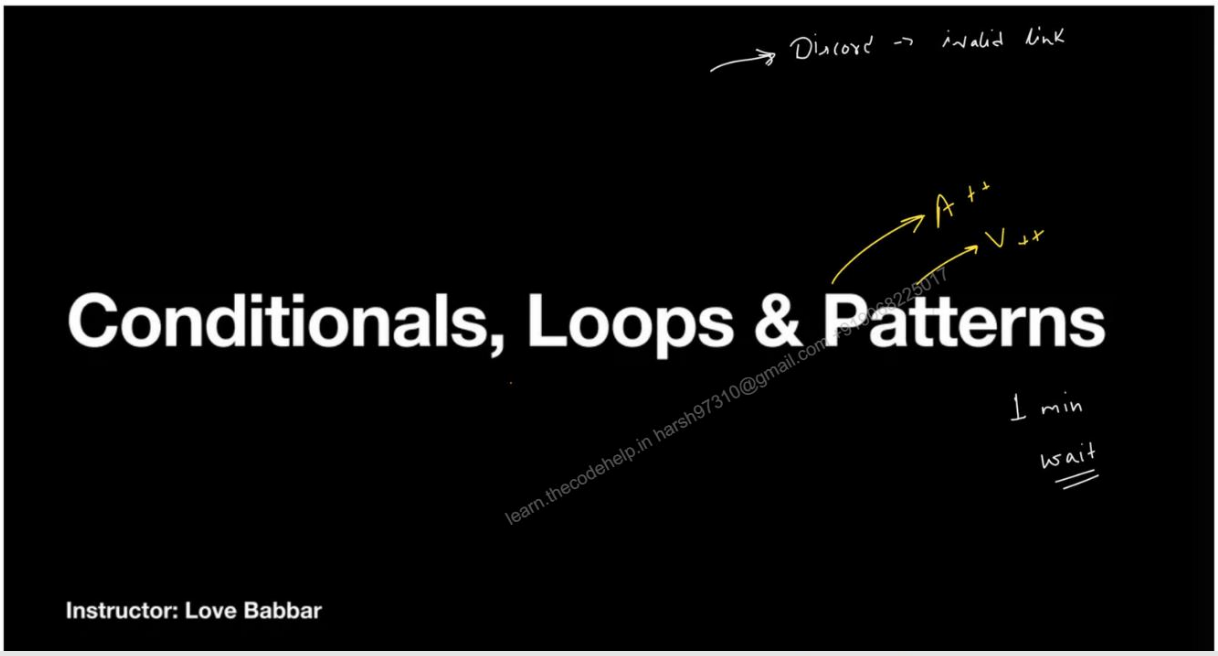
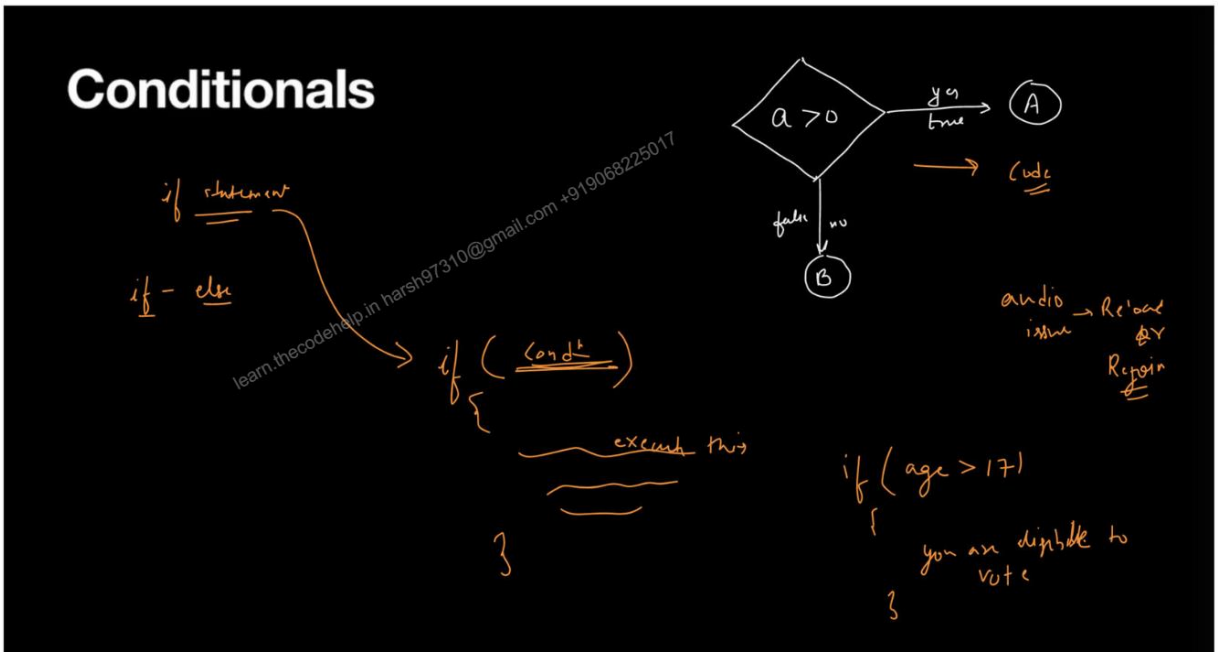
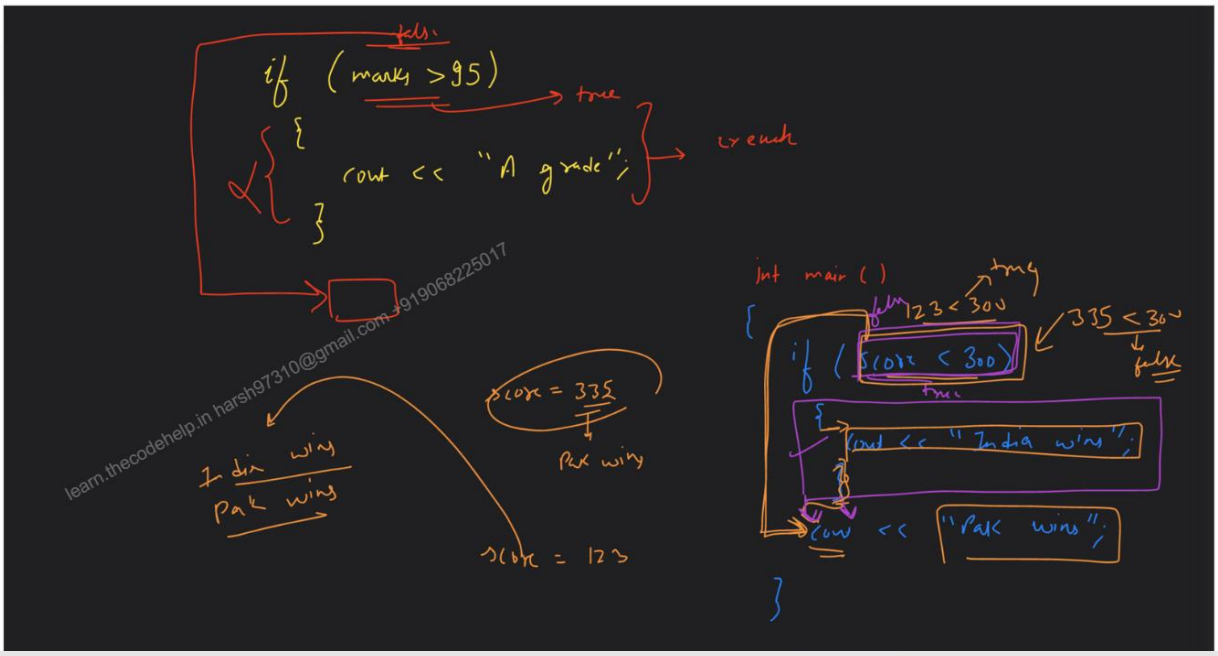
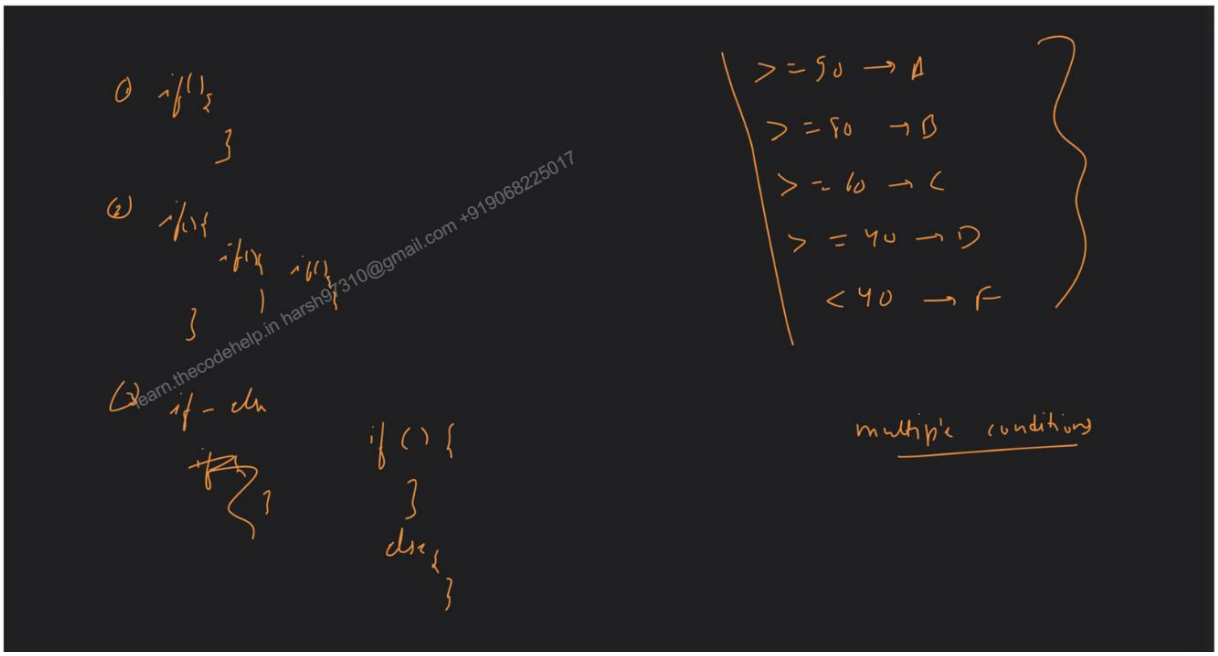
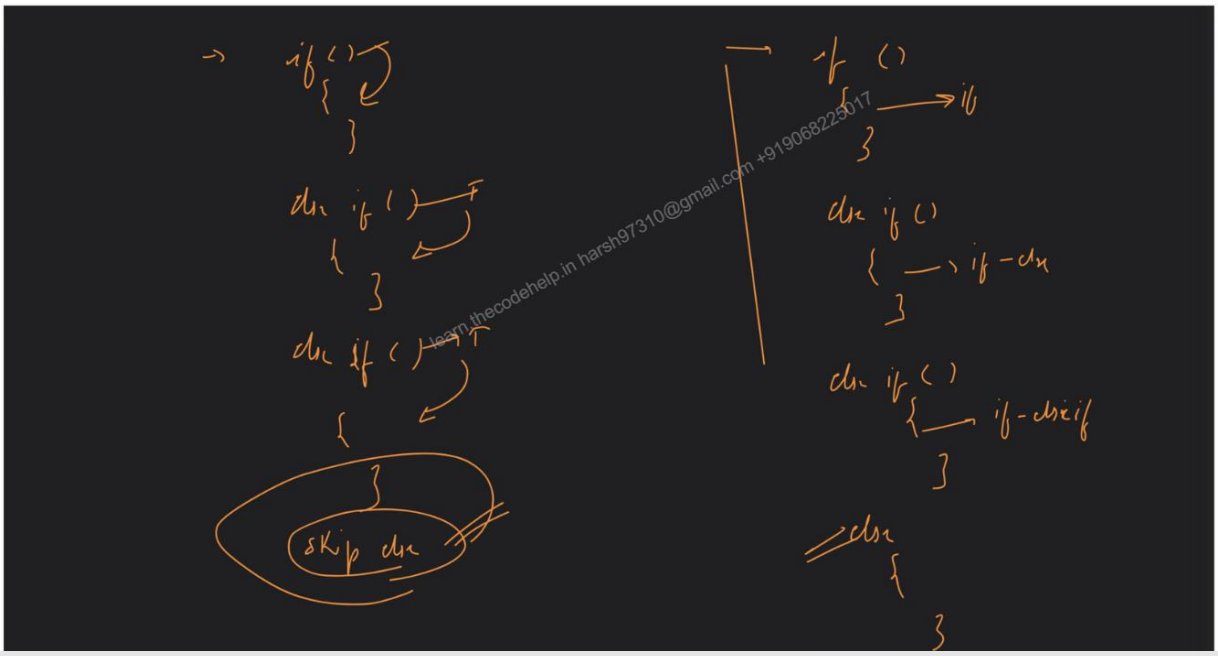
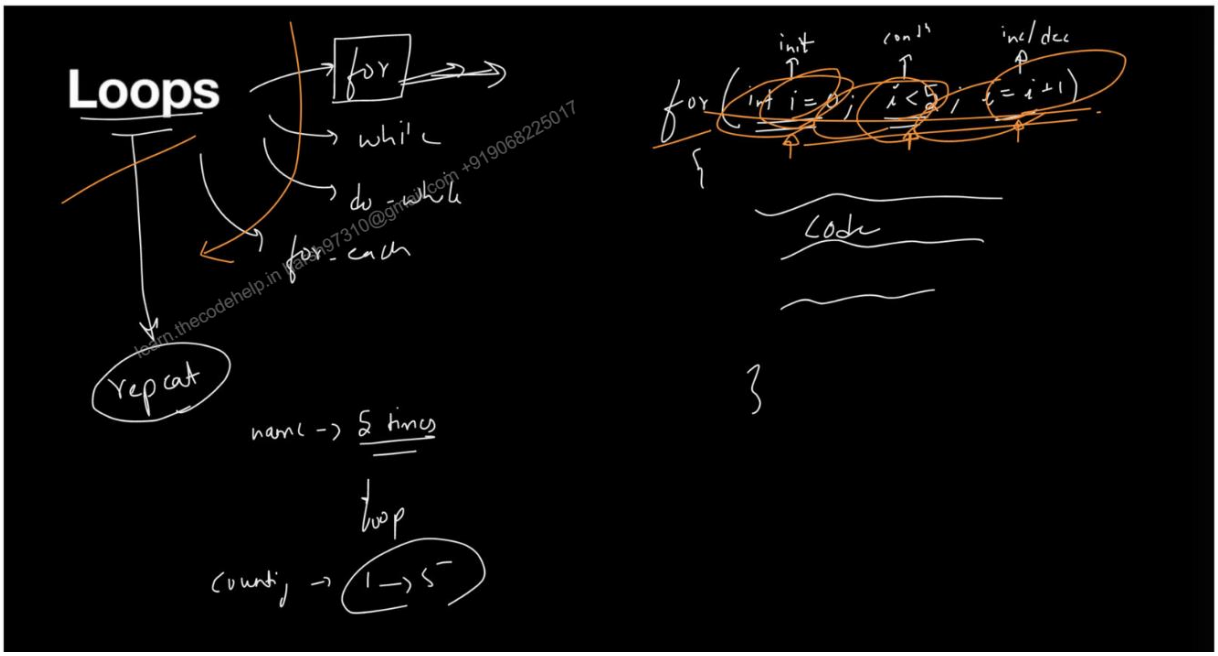


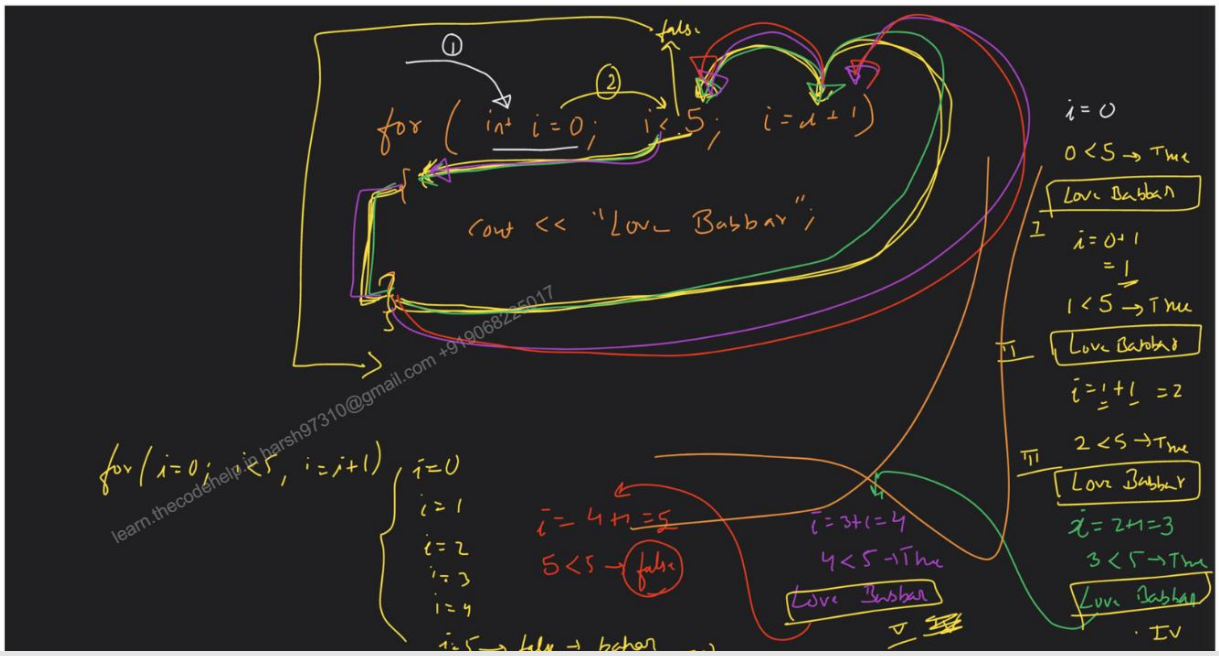


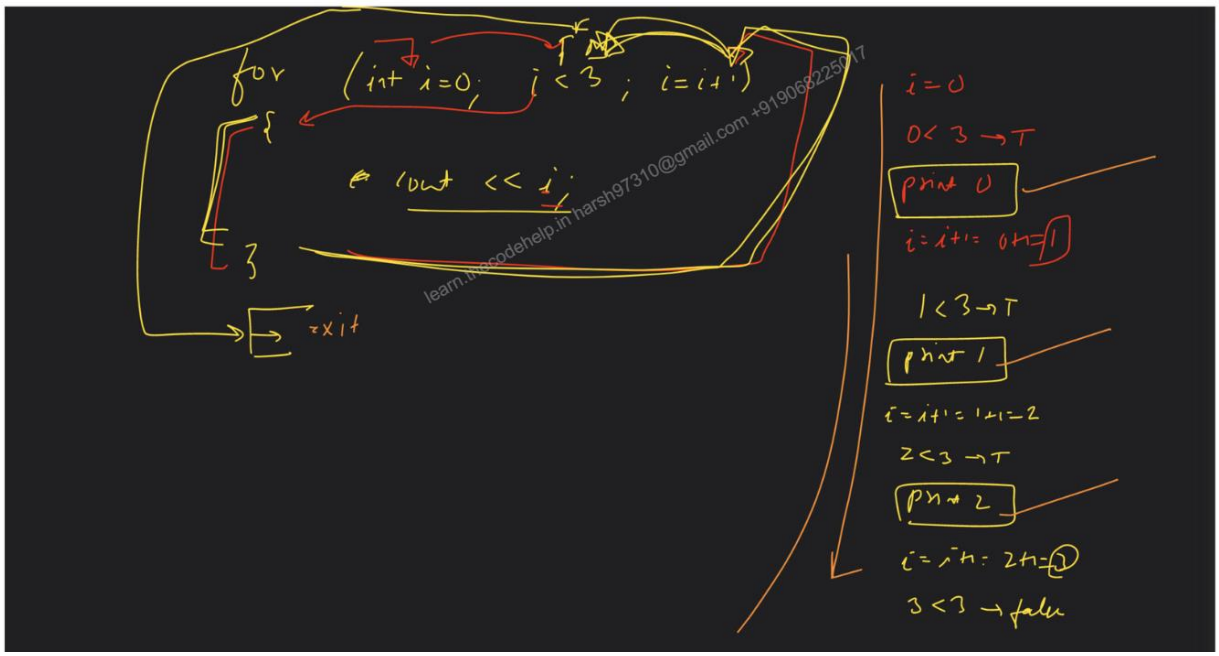


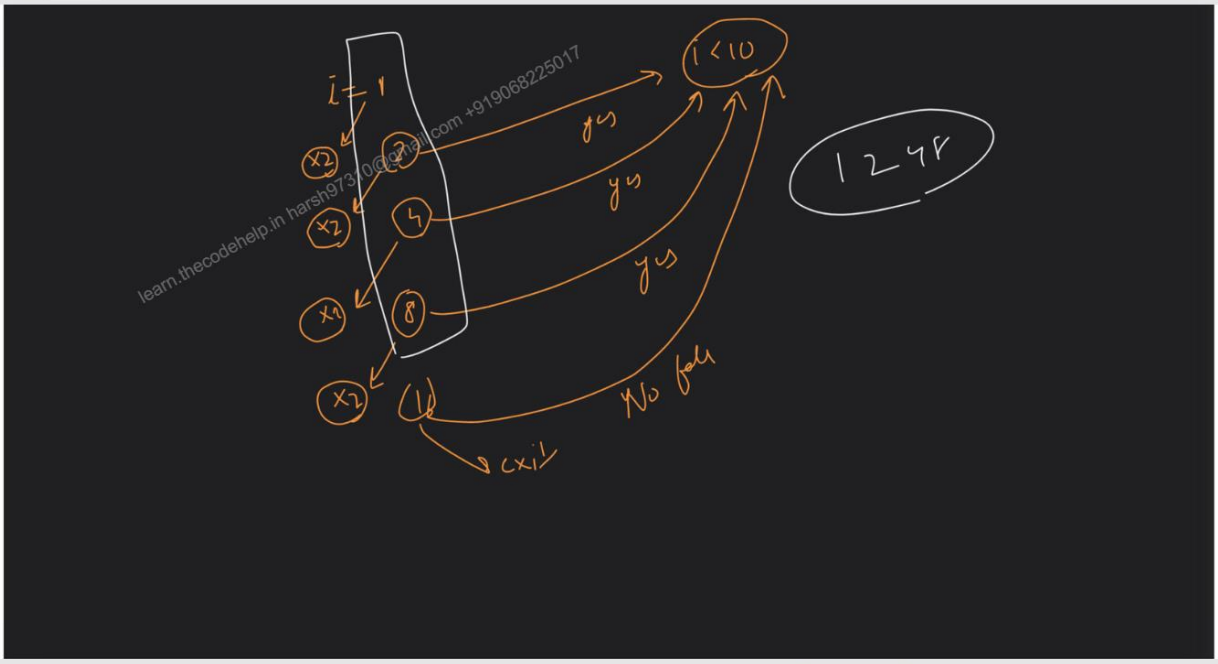


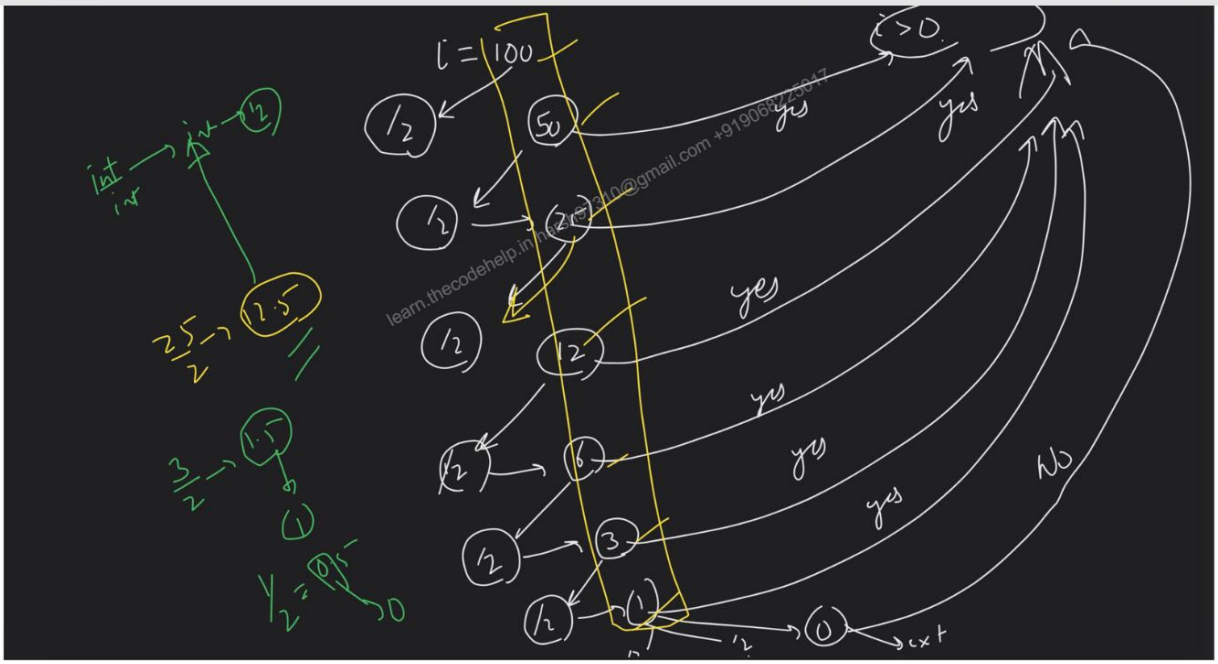


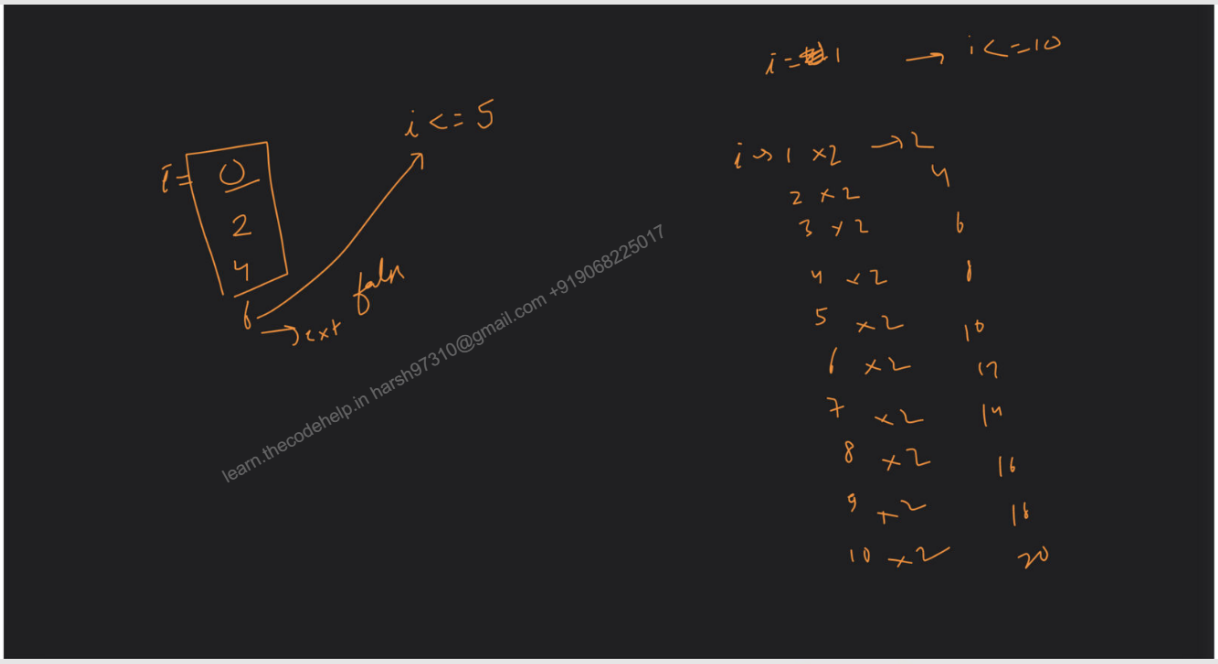


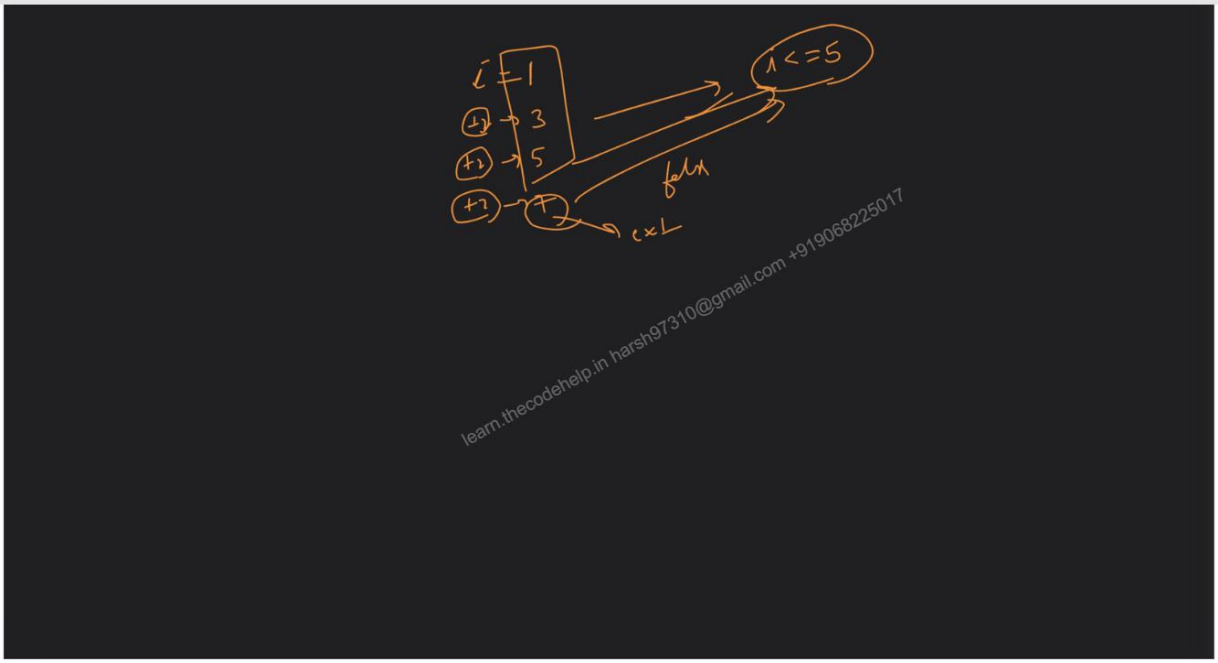

for (int i=0; i<5; i=i+1)
cout << "Love Babbar";
false
i=0
0<5 → True
Love Babbar
i=0+1 =1
1<5 → True
Love Babbar
i=1+1 =2
2<5 → True
Love Babbar
i=2+1=3
3<5 → True
Love Babbar
IV
i=3+1=4
4<5 → True
Love Babbar
i=4+1=5
5<5 → false
for(i=0; i<5; i=i+1)
i=0
i=1
i=2
i=3
i=4
learn.thecodehelp.in harsh97310@gmail.com +919068225017

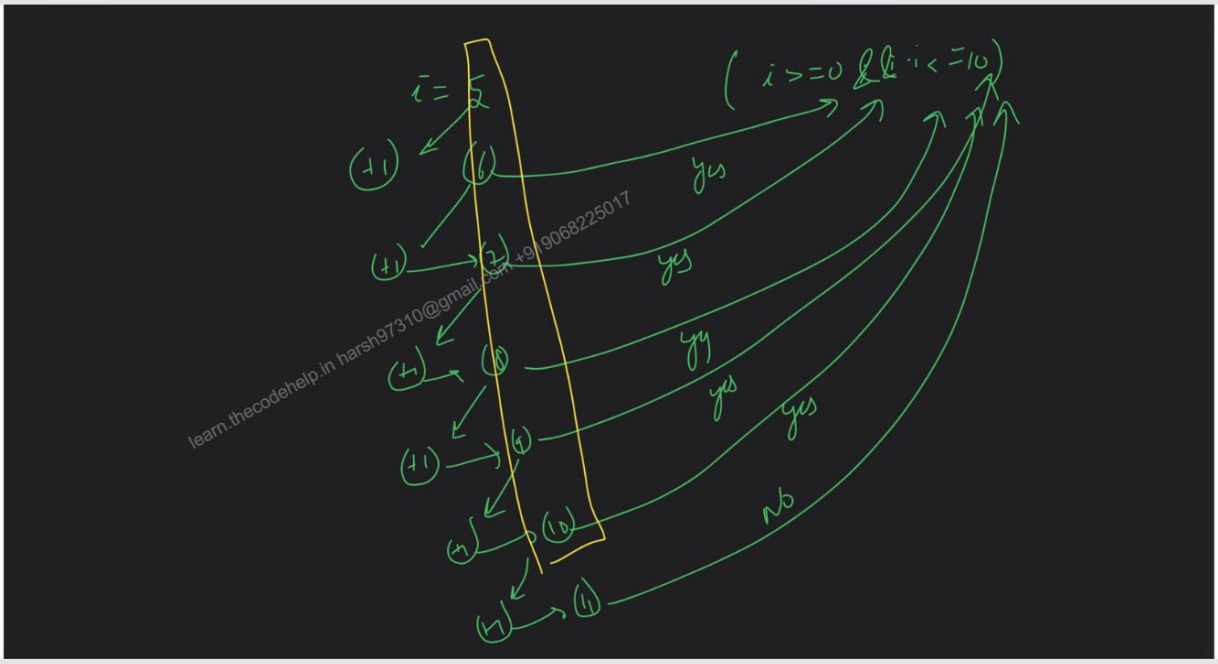

for (int i=0; i<3; i=i+1)
cout << i;
exit
i=0
0<3 → T
print 0
i=i+1= 0+1=1
1<3 → T
print 1
i=i+1= 1+1=2
2<3 → T
print 2
i=i+1= 2+1=3
3<3 → false
learn.thecodehelp.in harsh97310@gmail.com +919068225017

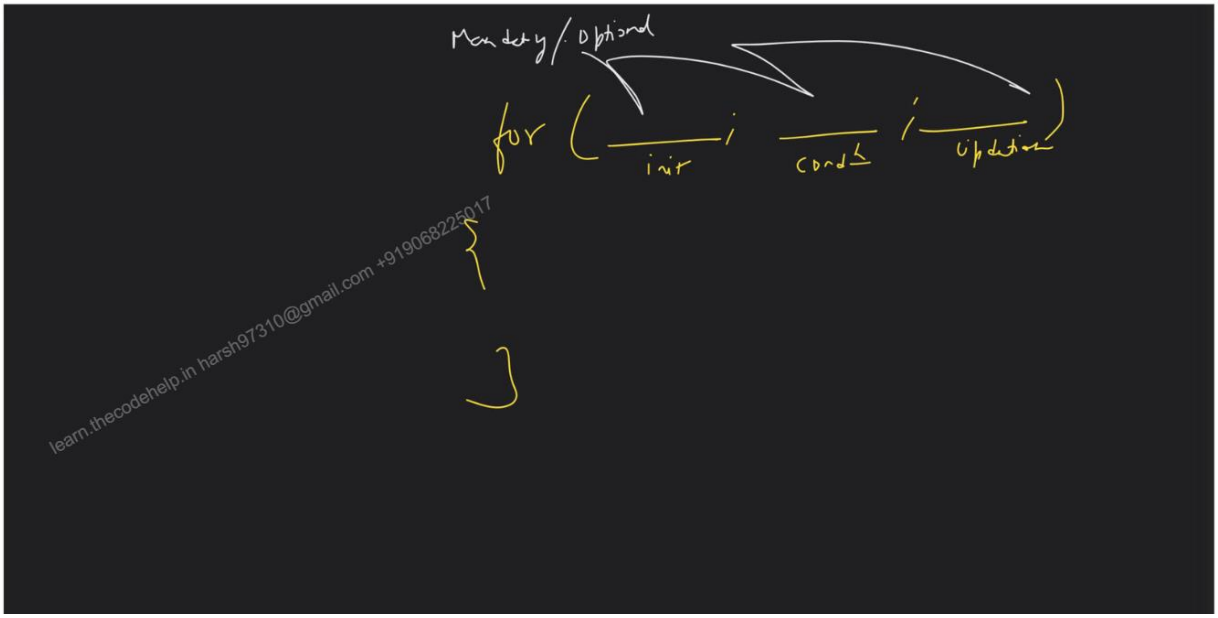


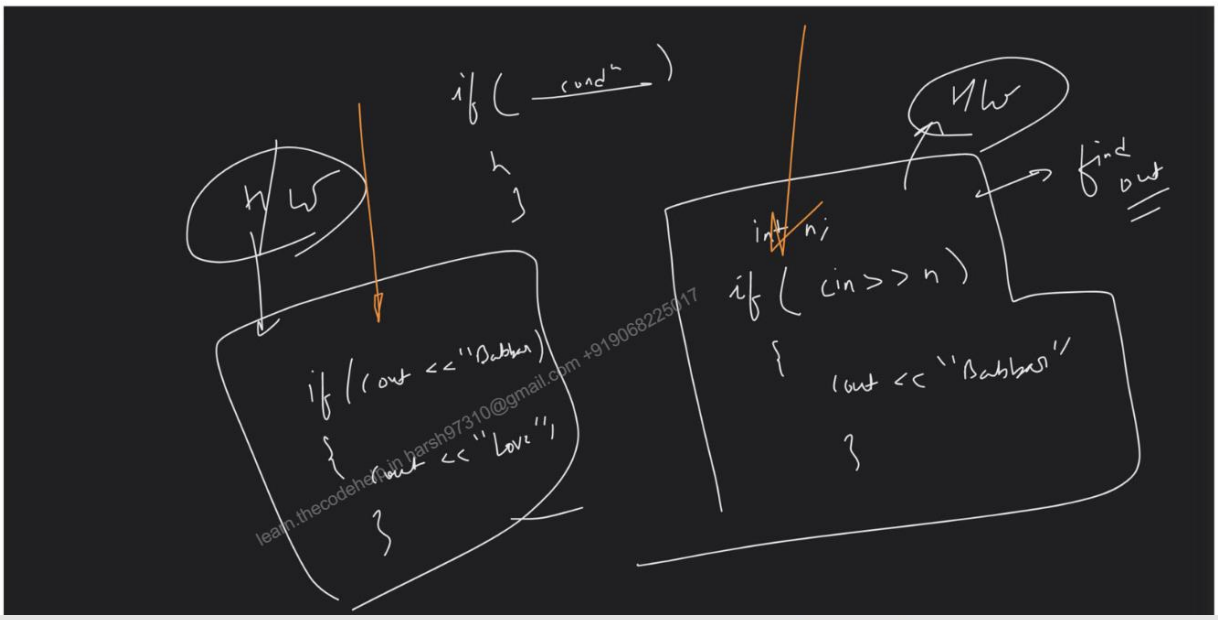


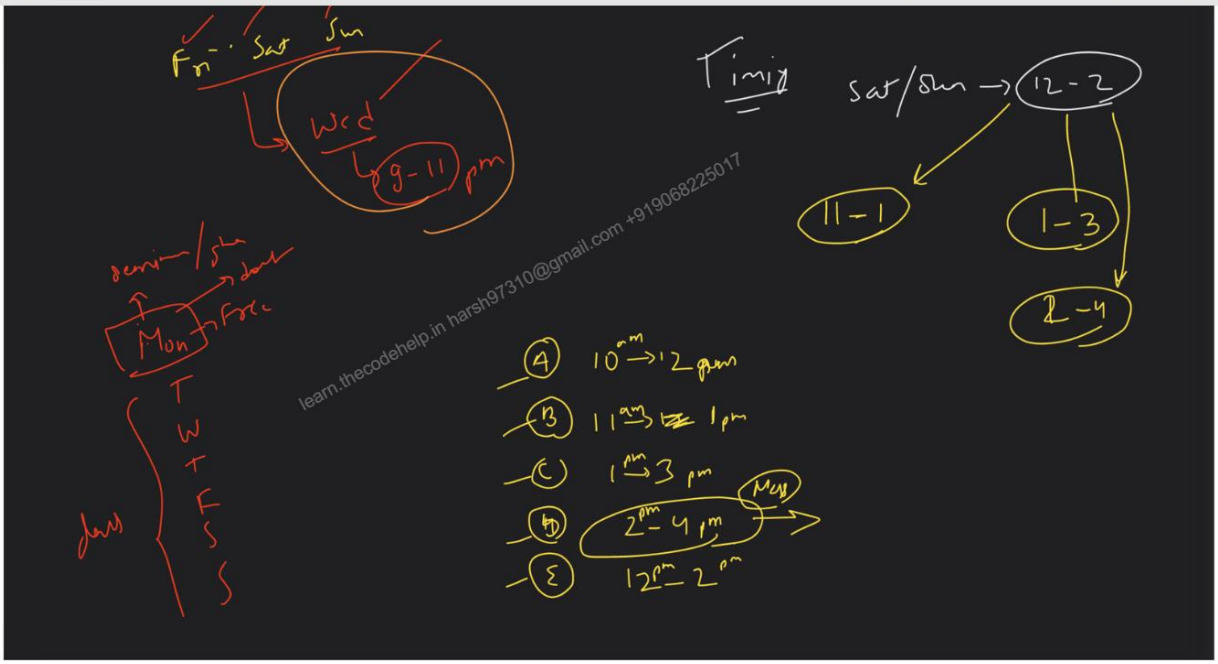


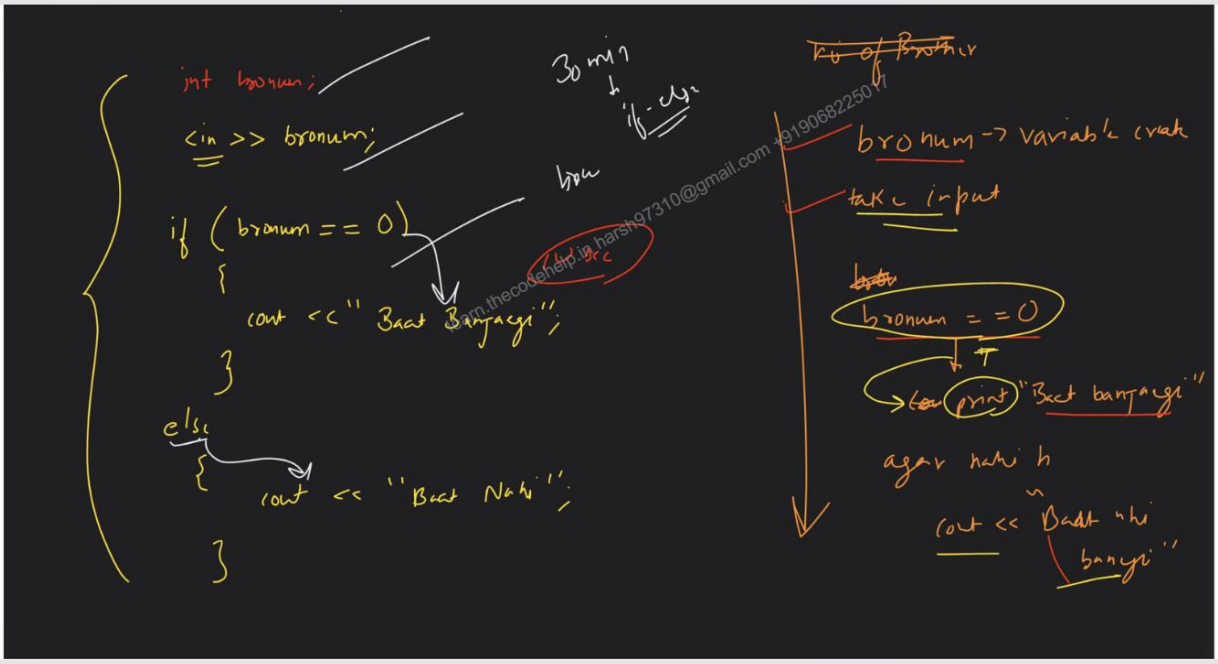


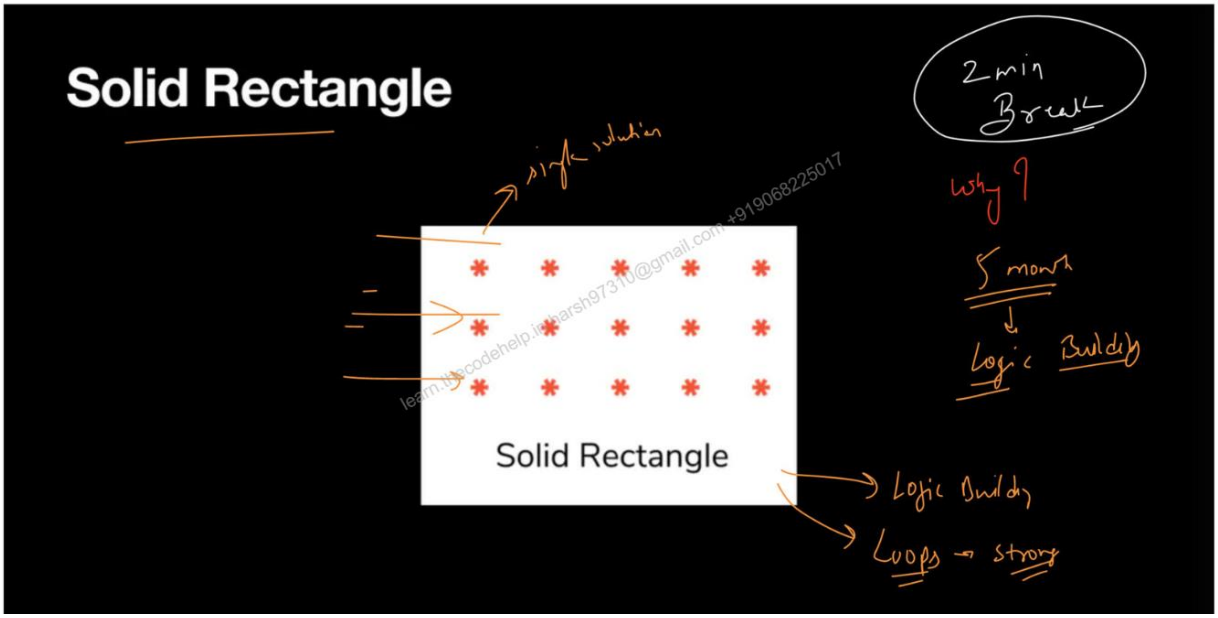


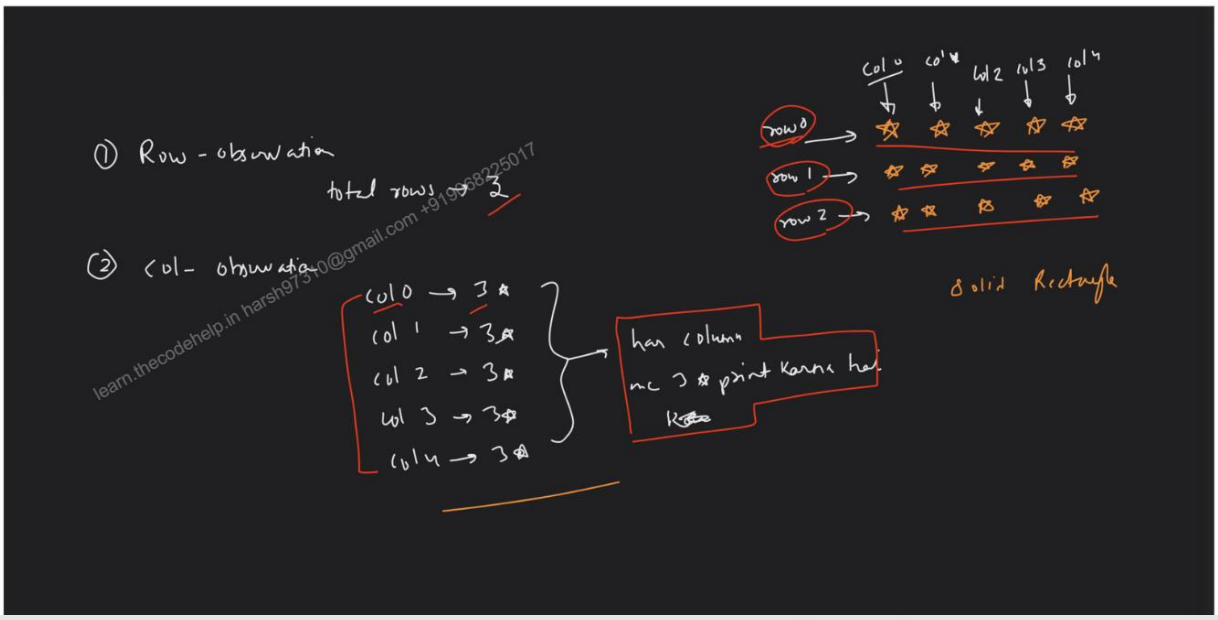


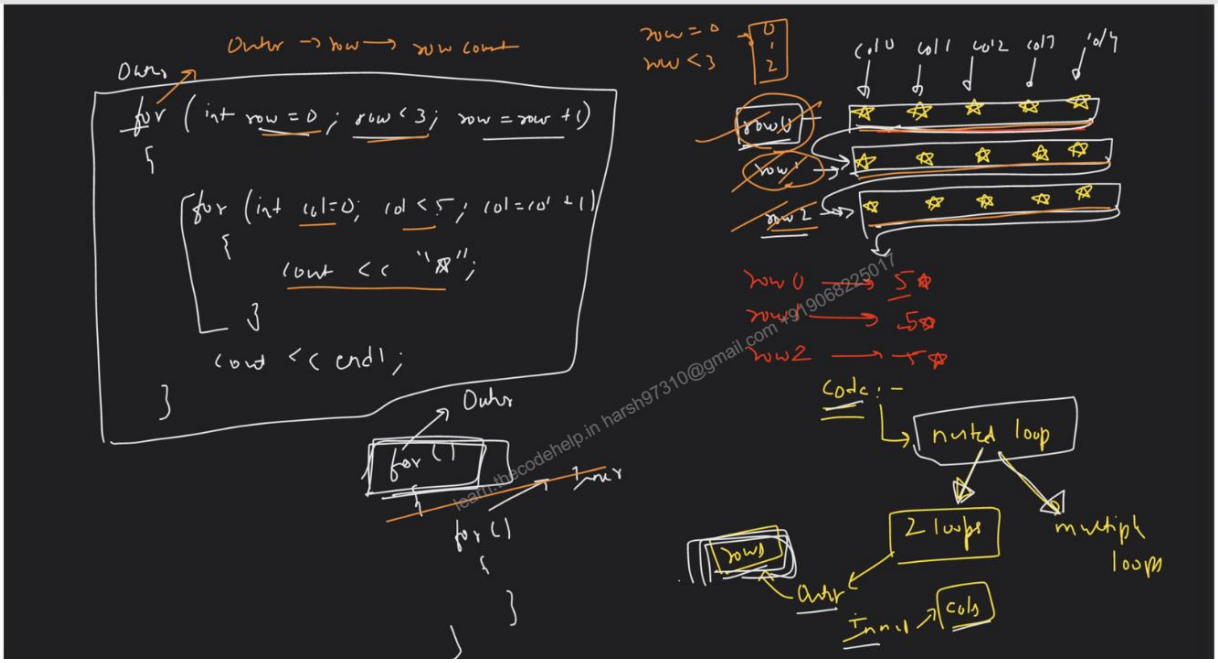


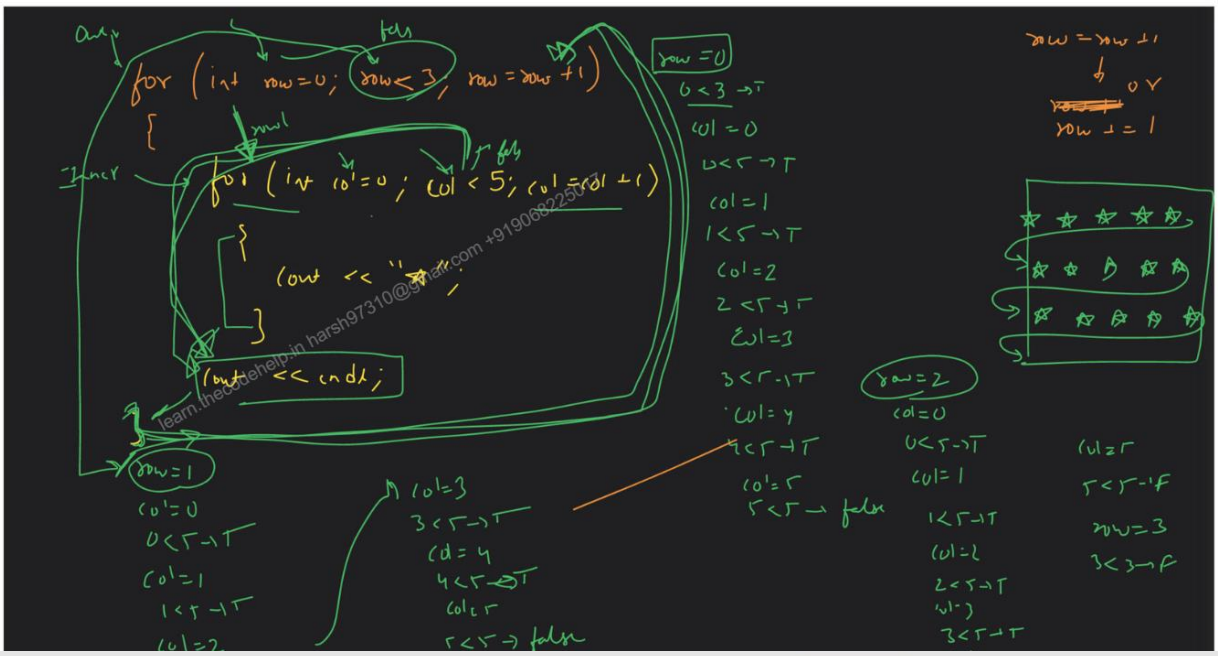


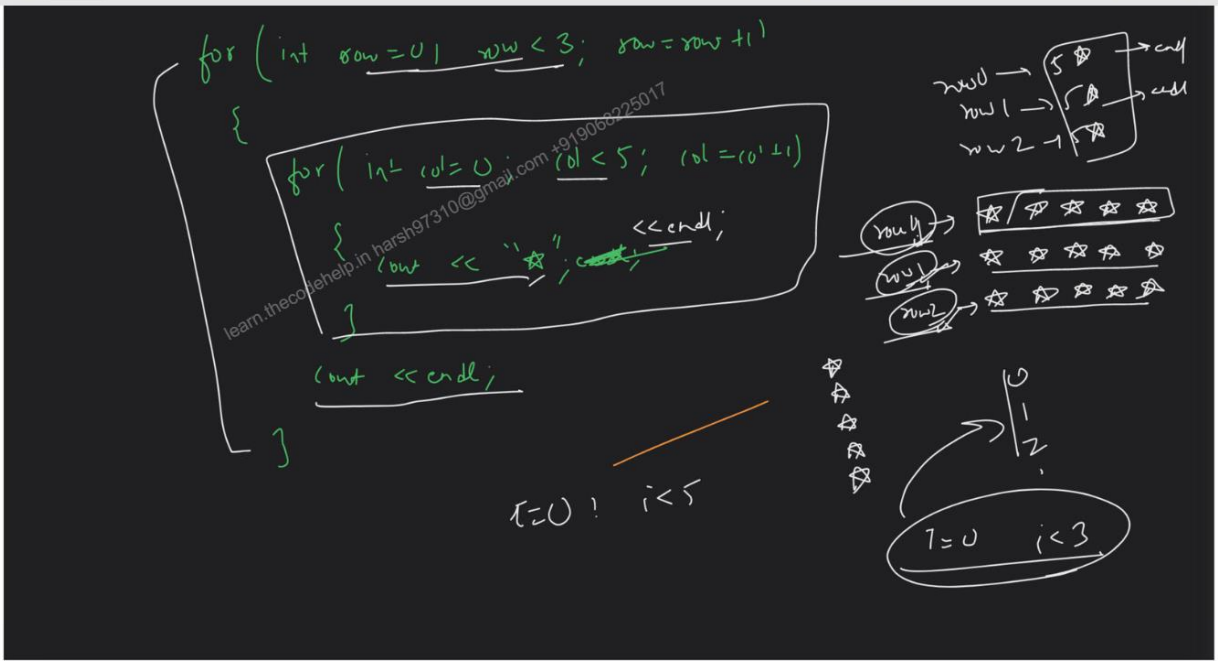

int bronum;
cin >> bronum;
if (bronum == 0)
{
cout << "Baat Banyagi";
}
else
{
cout << "Baat Nahi";
}
bronum -> variable create
take input
bronum == 0
print "Baat banyagi"
agar nahi h
cout << Baat nhi banyagi

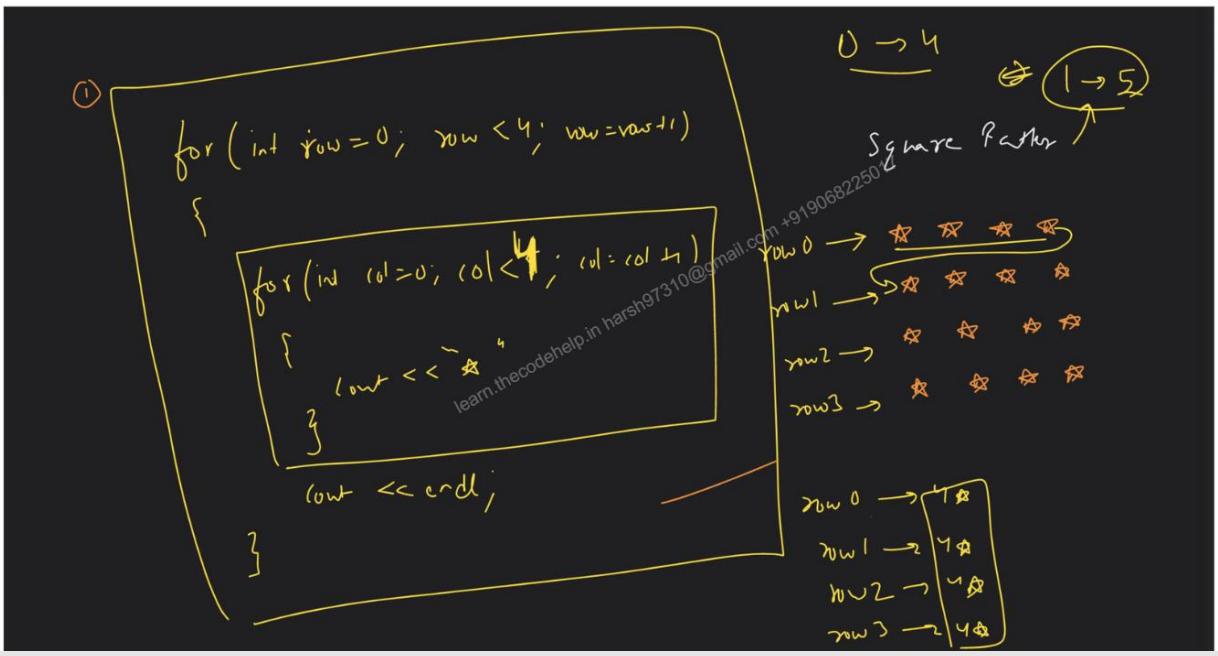

Solid Rectangle
2 min Break
single solution
Solid Rectangle
Why ?
5 month
Logic Building
Logic Building
Loops → strong

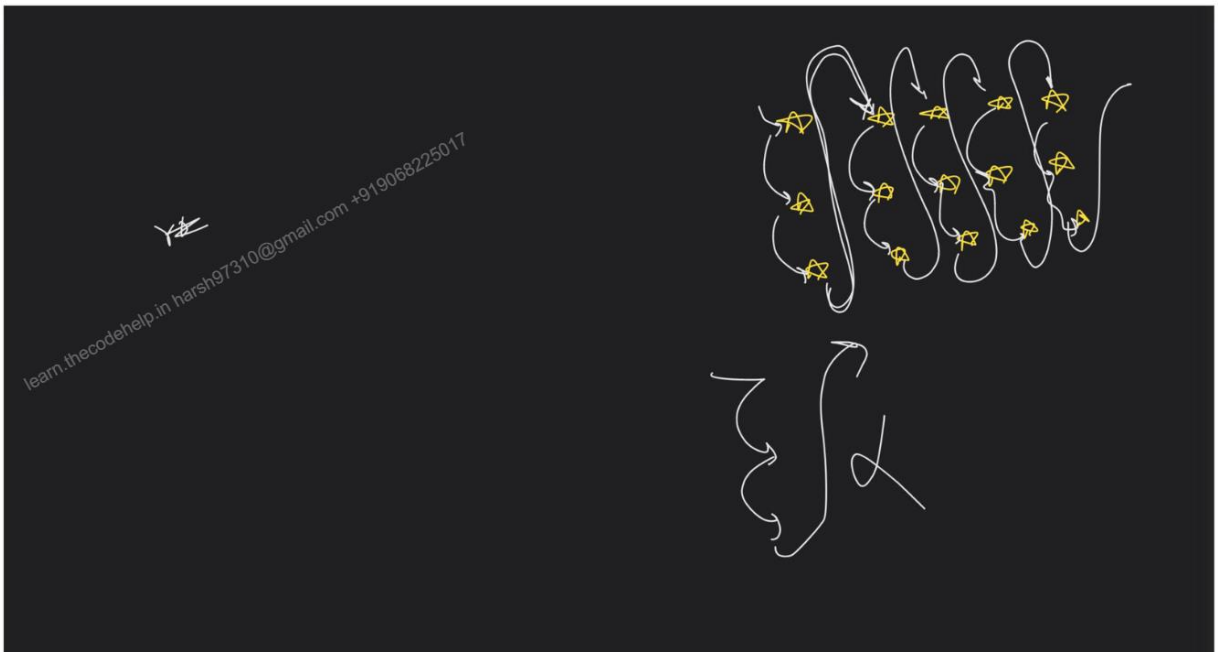


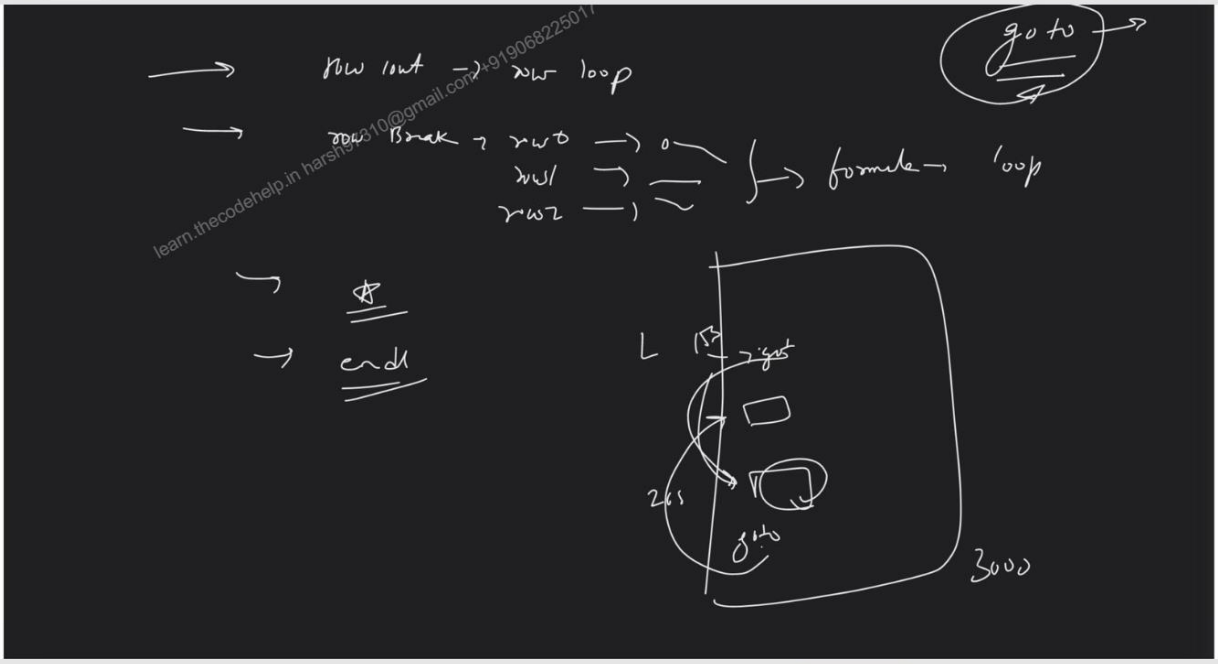


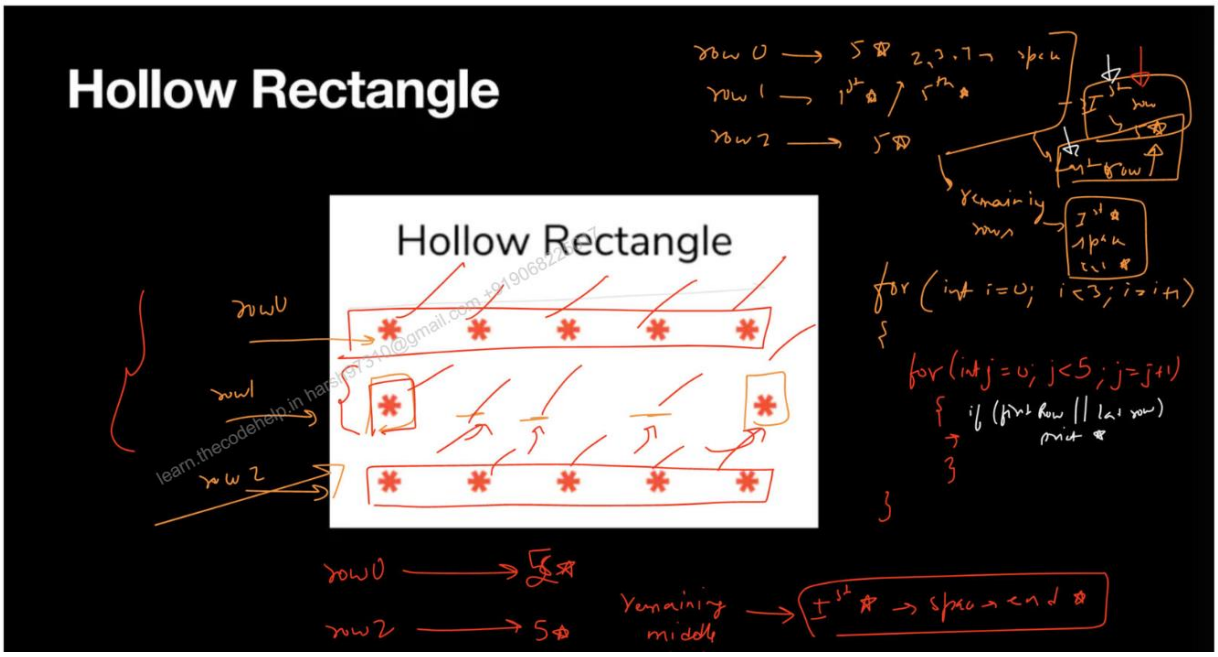

for (int row=0; row<3; row=row+1)
for (int col=0; col<5; col=col+1)
cout << "*";
cout << endl;
row=0
row=1
row=2
learn.thecodehelp.in harsh97310@gmail.com +919066225017

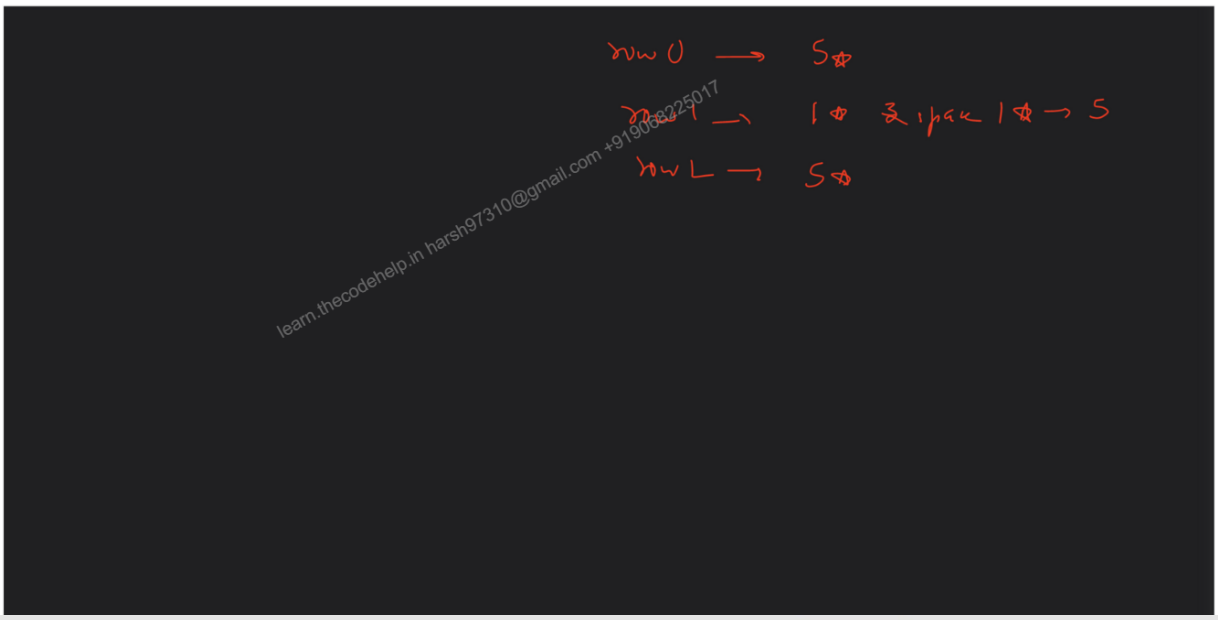

for (int row=0; row<3; row=row+1)
for (int col=0; col<5; col=col+1)
cout << "*";
cout << endl;
learn.thecodehelp.in harsh97310@gmail.com +919066225017
i=0 ; i<5
i=0 ; i<3

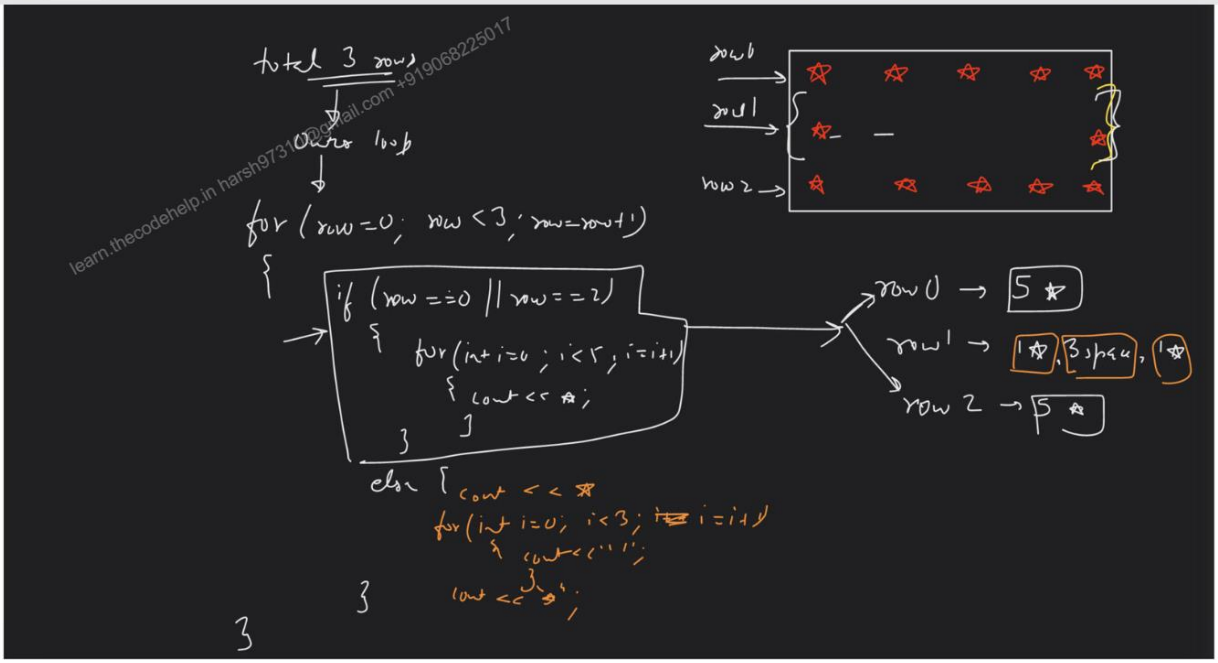

0 → 4
1 → 5
Square Pattern
for (int row = 0; row < 4; row = row + 1)
{
for (int col = 0; col < 4; col = col + 1)
{
cout << "*";
}
cout << endl;
}
row 0
row 1
row 2
row 3
learn.thecodehelp.in harsh97310@gmail.com +919068225017

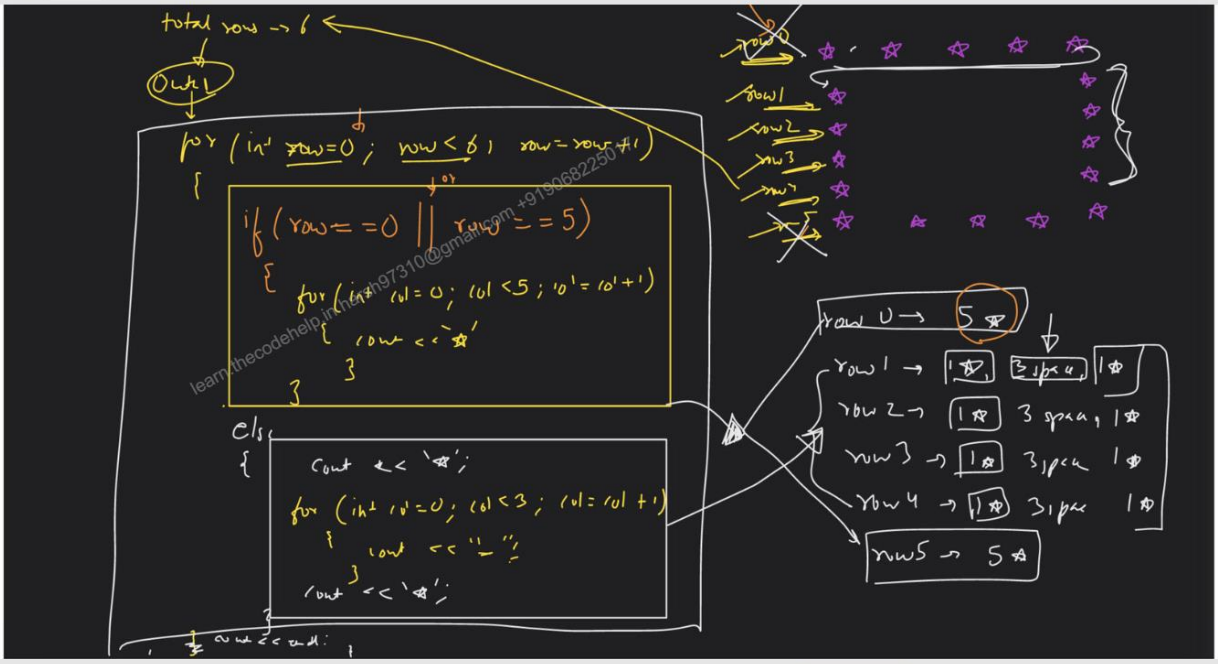

learn.thecodehelp.in harsh97310@gmail.com +919068225017

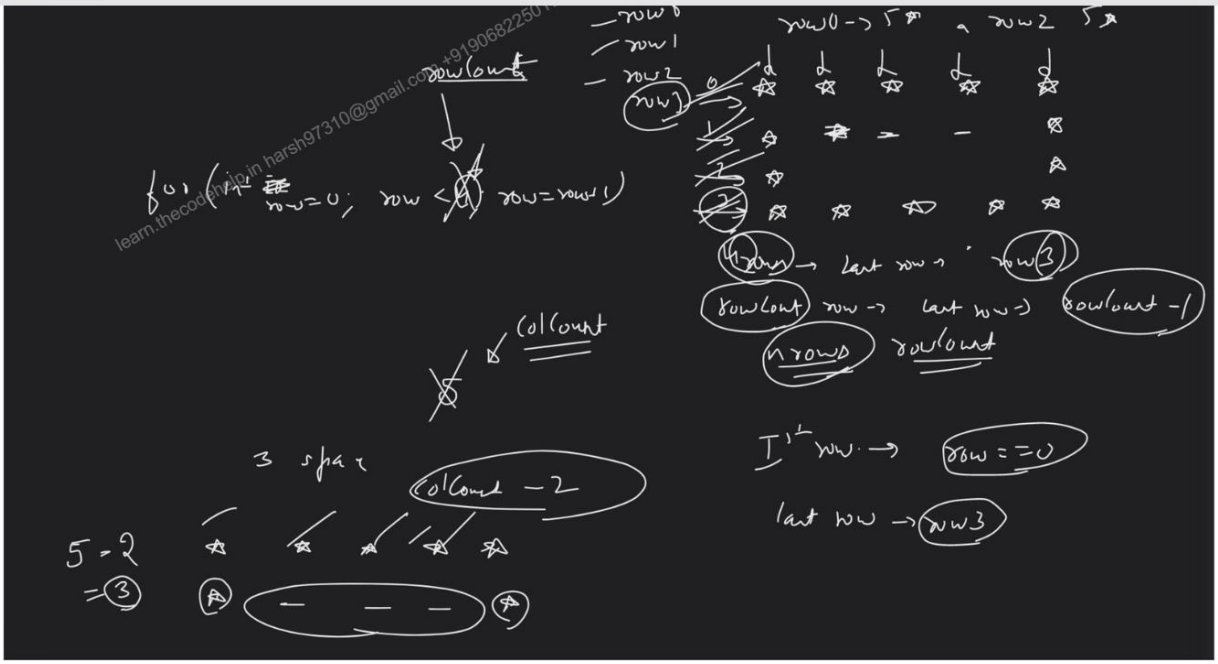

row int → row loop
row Break → row 0
row 1
row 2
formula → loop
go to
star
endl
L
right
left
3000

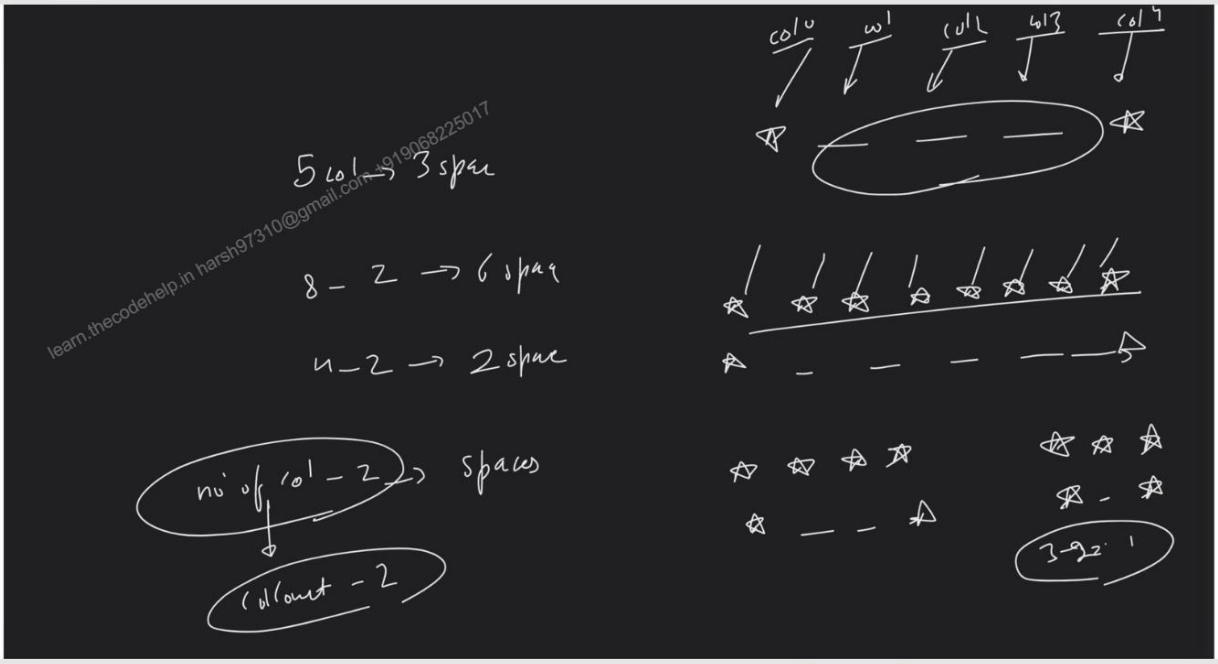

Hollow Rectangle
row 0 → 5 ★
row 1 → 1st ★ / 5th ★
row 2 → 5 ★
2,3,4 → space
remaining rows
for (int i=0; i<3; i=i+1)
{
for (int j=0; j<5; j=j+1)
{ if (first row || last row)
print ★
}
}
Hollow Rectangle
row 0
row 1
row 2
row 0 → 5 ★
row 2 → 5 ★
Remaining middle → 1st ★ → space → end ★

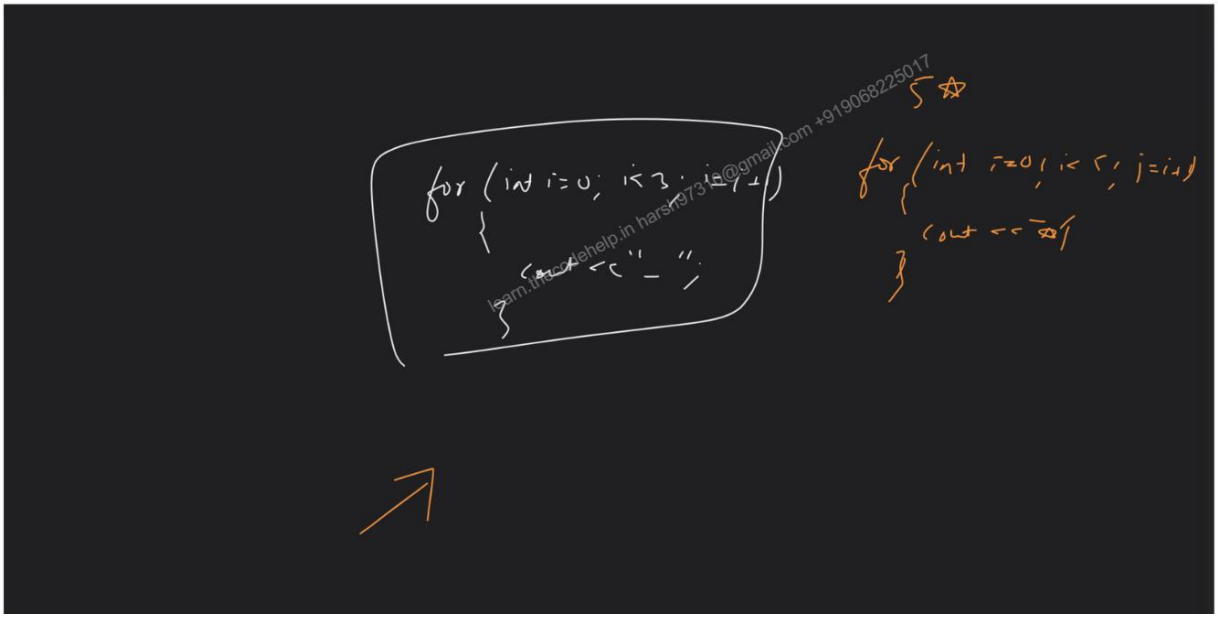


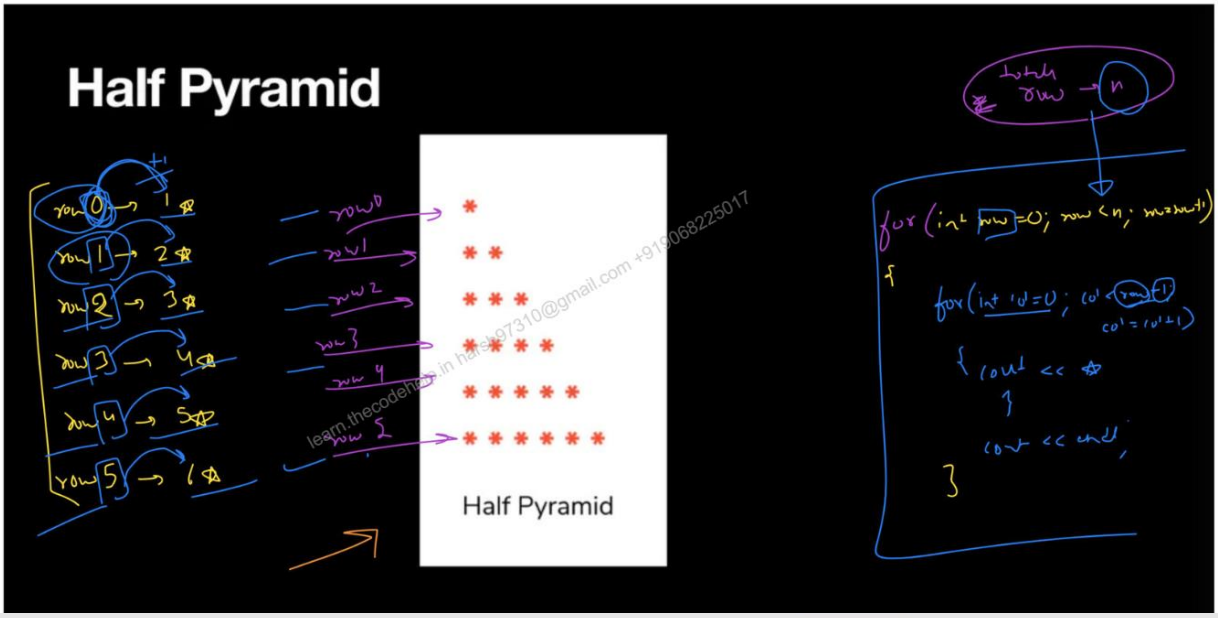


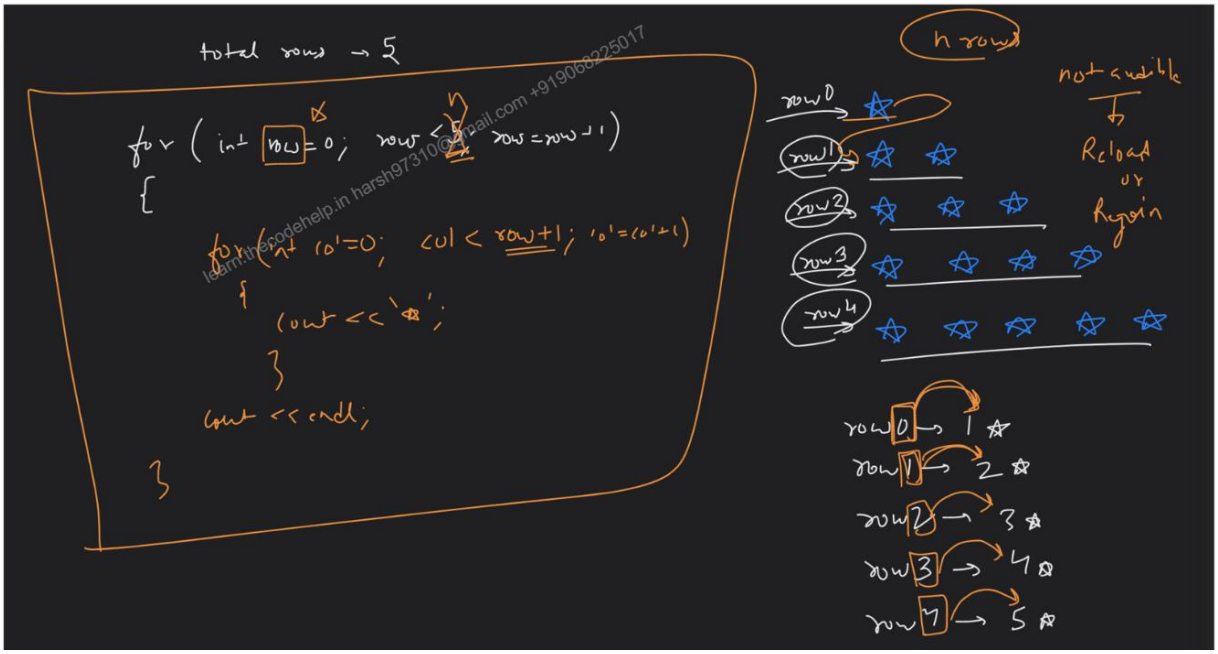

total row -> 6
Output
for (int row=0; row<6; row=row+1)
if (row==0 || row==5)
for (int col=0; col<5; col=col+1)
cout << '*'
else
cout << '*';
for (int col=0; col<3; col=col+1)
cout << ' ';
cout << '*';
row 0 -> 5
row 1 -> 1 3 spaces 1
row 2 -> 1 3 spaces 1
row 3 -> 1 3 spaces 1
row 4 -> 1 3 spaces 1
row 5 -> 5

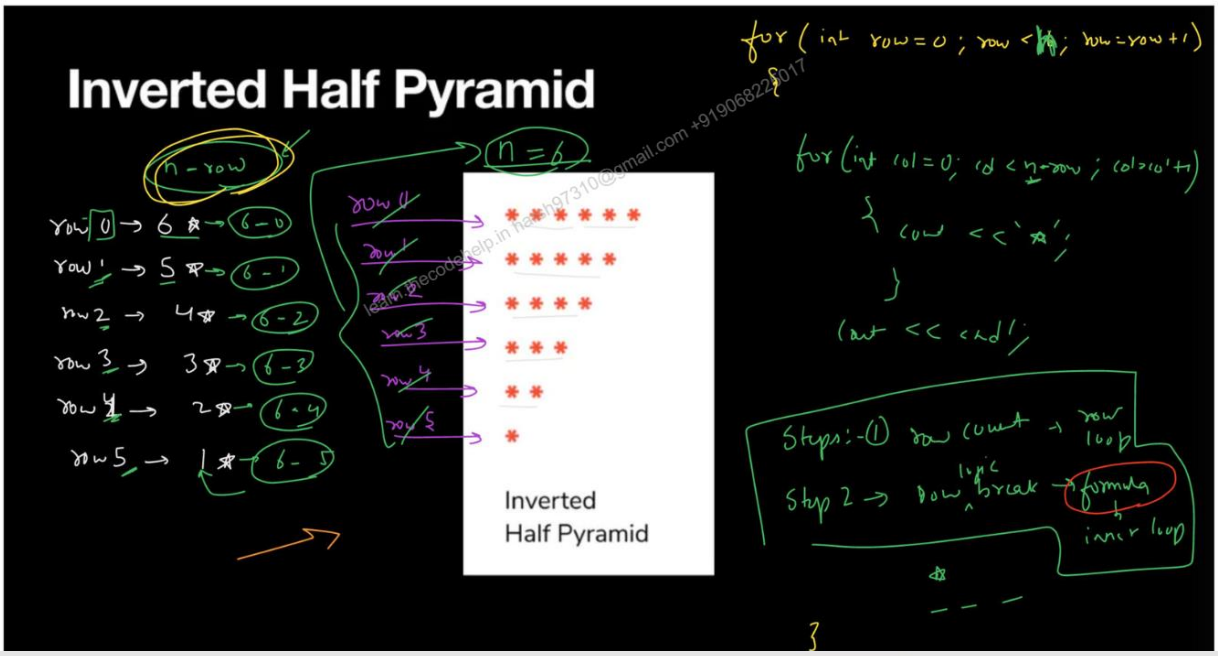

rowCount
for (int row=0; row<rowCount; row=row+1)
colCount
3 spaces
colCount - 2
5-2
=3
1st row -> row==0
last row -> row3
rowCount row -> last row -> rowCount-1
nrows rowCount

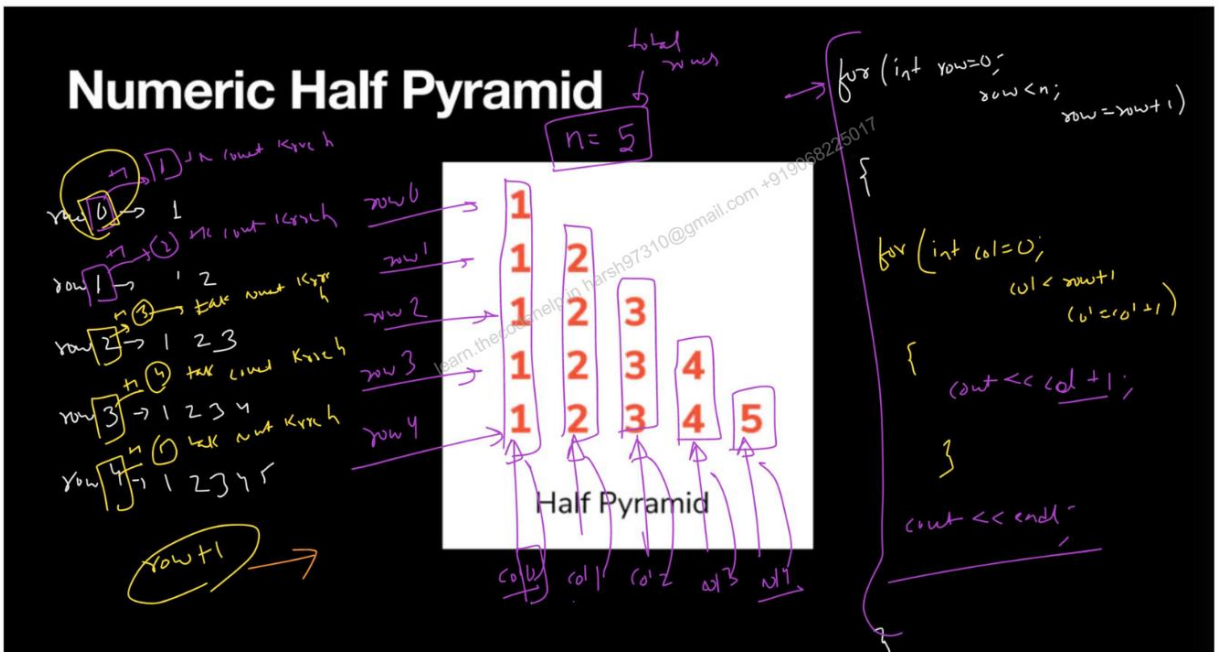


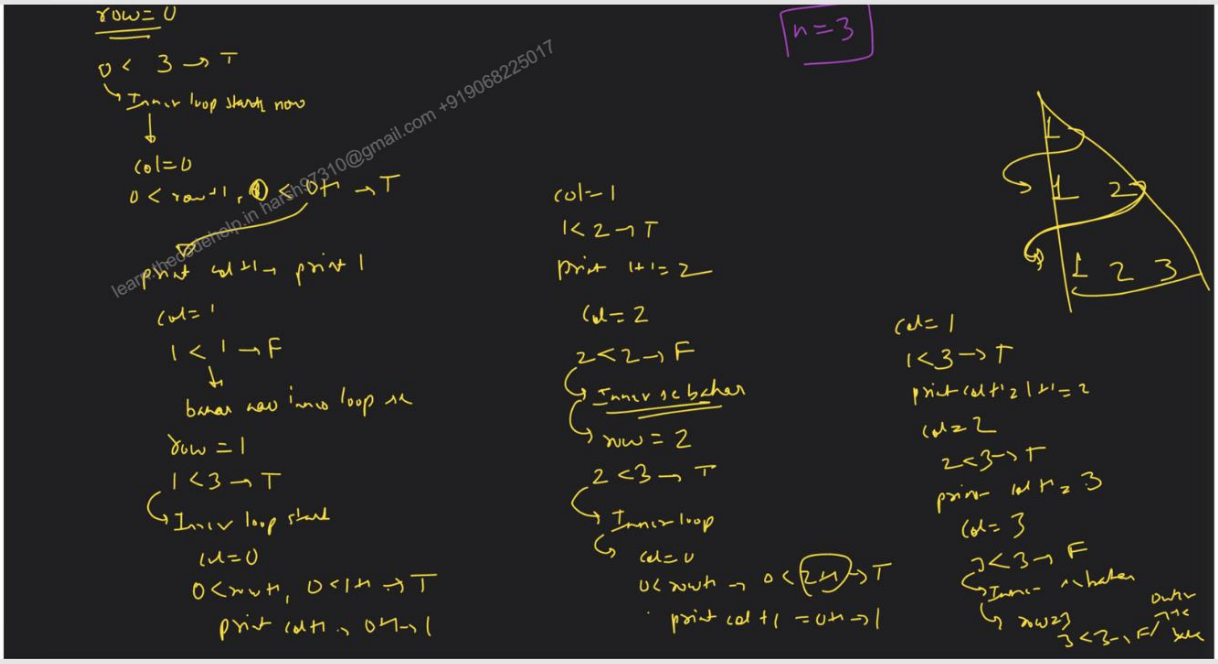


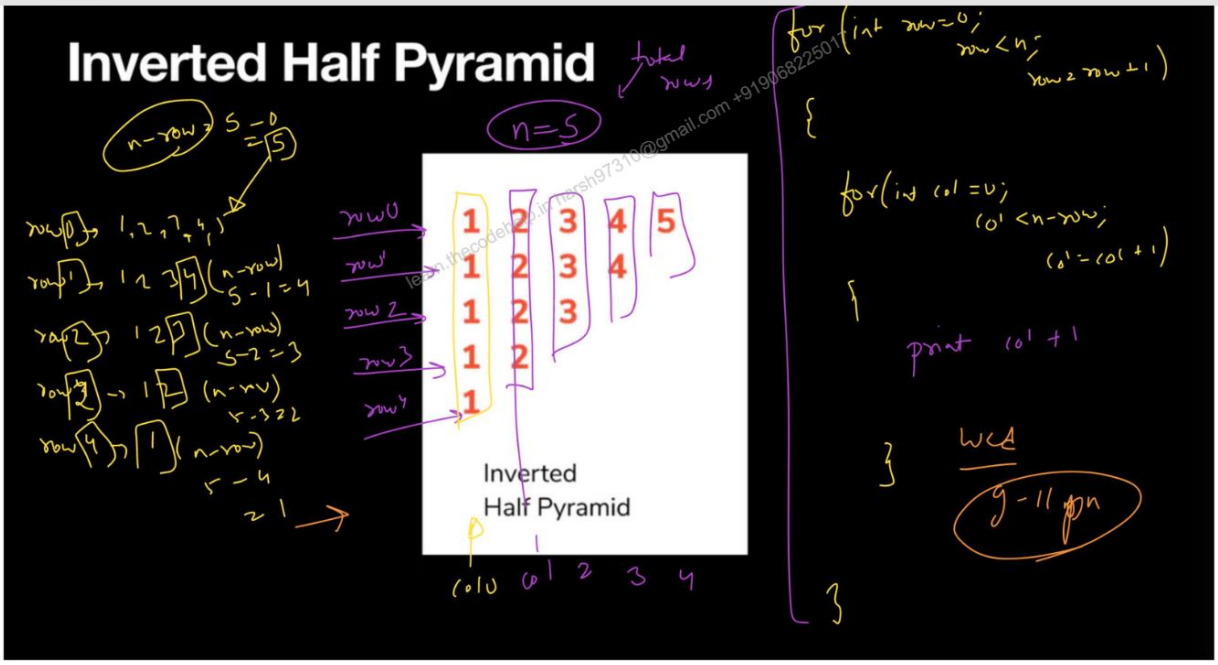

Half Pyramid
Half Pyramid
total row → n
for (int row=0; row<n; row++)
{
for (int col=0; col<row+1; col=col+1)
{ cout << *
}
cout << endl;
}
learn.thecodehelp.in harsh97310@gmail.com +919068225017


total rows → 5
for (int row=0; row<5; row=row+1)
{
for(int col=0; col<row+1; col=col+1)
{
cout << '*';
}
cout << endl;
}
n rows
not visible
Reload
or
Region
learn.thecodehelp.in harsh97310@gmail.com +919068225017

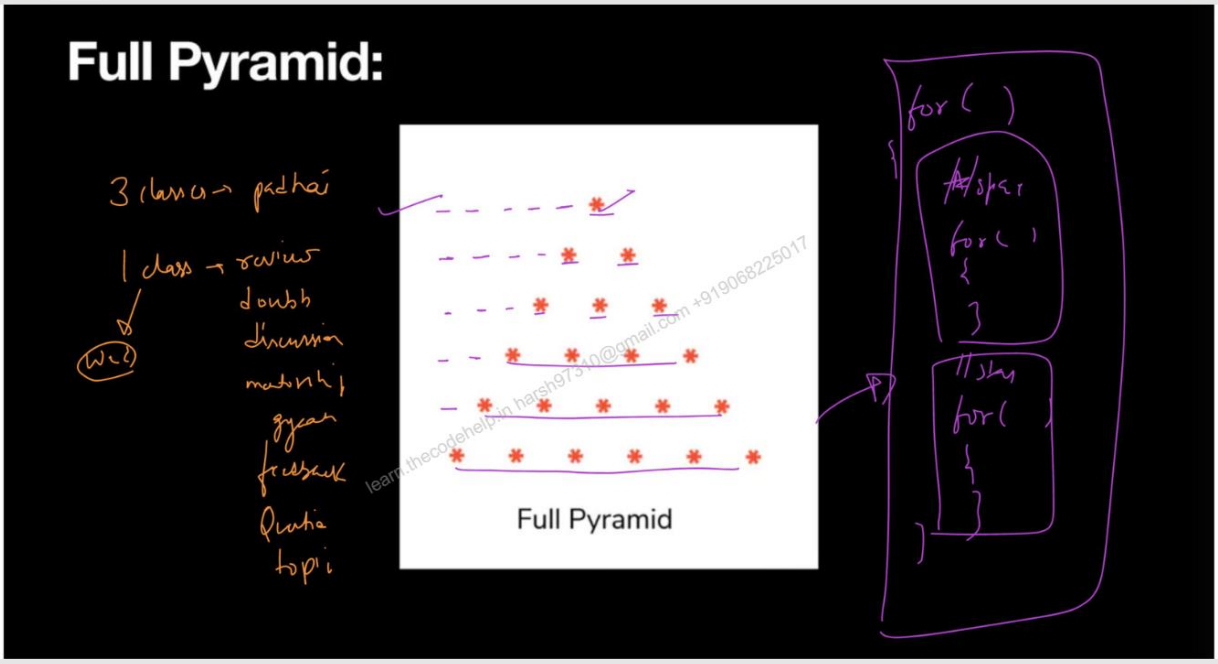

Inverted Half Pyramid
Inverted
Half Pyramid
Steps:-
Step 2 →


Numeric Half Pyramid
n= 5
Half Pyramid

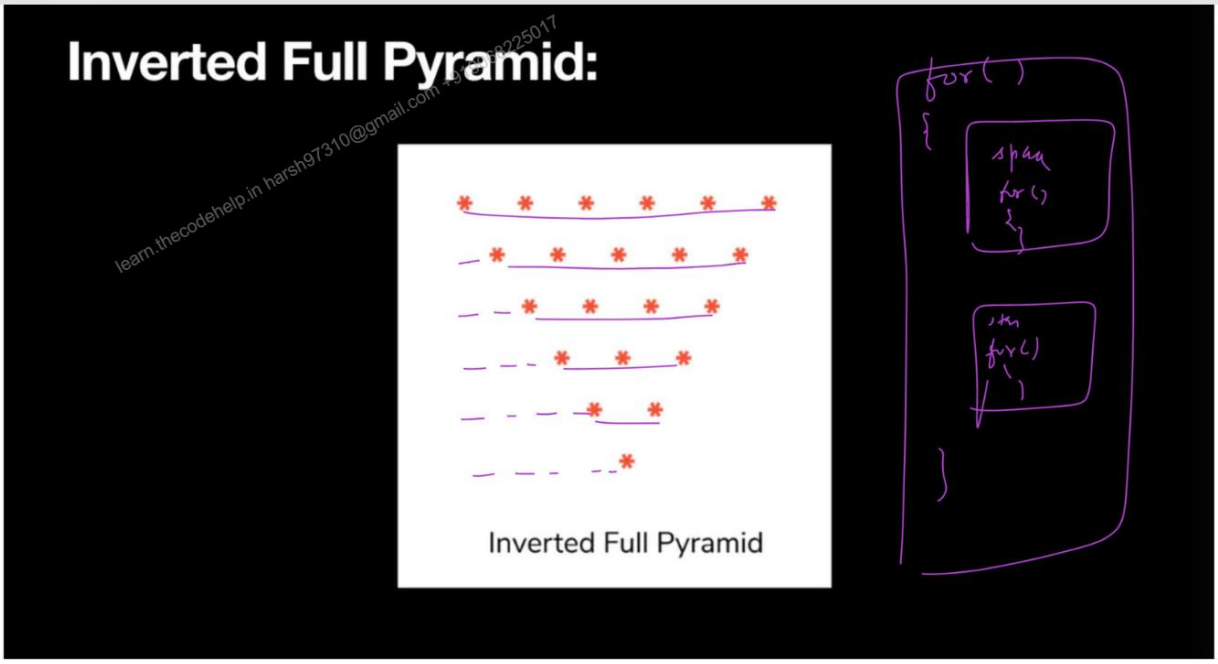

row = 0
0 < 3 → T
Inner loop starts now
col = 0
0 < row+1, 0 < 0+1 → T
print col+1 → print 1
col = 1
1 < 1 → F
break now inner loop se
row = 1
1 < 3 → T
Inner loop start
col = 0
0 < row+1, 0 < 1+1 → T
print col+1 → 0+1 → 1
col = 1
1 < 2 → T
print 1+1 = 2
col = 2
2 < 2 → F
Inner se bahar
row = 2
2 < 3 → T
Inner loop
col = 0
0 < row+1 → 0 < 2+1 → T
print col+1 = 0+1 → 1
n = 3
1
1 2
1 2 3
col = 1
1 < 3 → T
print col+2-1+1 = 2
col = 2
2 < 3 → T
print col+2 = 3
col = 3
3 < 3 → F
Inner se bahar
row++
3 < 3 → F
Outer se bahar


Inverted Half Pyramid
total rows
n=5
n-row
s = 0
s = 5
row0 → 1,2,3,4,5
row1 → 1 2 3 4 (n-row) 5-1=4
row2 → 1 2 3 (n-row) 5-2=3
row3 → 1 2 (n-row) 5-3=2
row4 → 1 (n-row) 5-4=1
row0
row1
row2
row3
row4
1 2 3 4 5
1 2 3 4
1 2 3
1 2
1
Inverted
Half Pyramid
col0 col1 2 3 4
for(int row=0; row<n; row=row+1)
{
for(int col=0; col<n-row; col=col+1)
{
print col+1
}
}
LWCA
9-11 pm

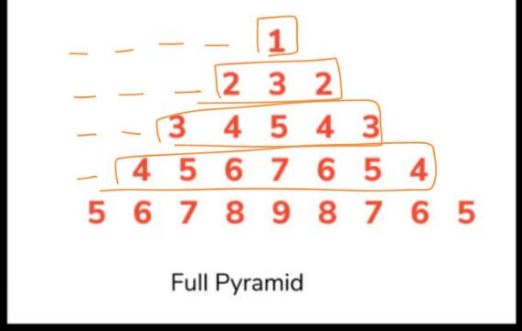
Full Pyramid:



Inverted Full Pyramid:

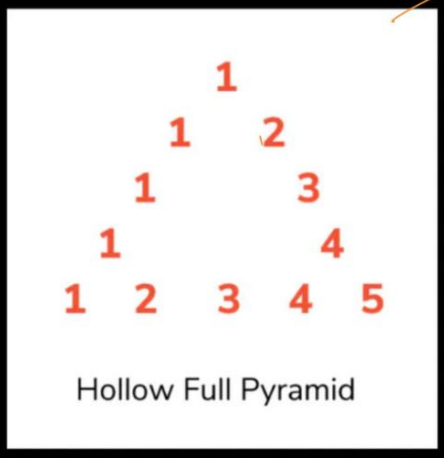


Numeric Full Pyramid:



Full Pyramid

Numeric Hollow Full Pyramid



Hollow Full Pyramid