Week-2 Lecture No 4

Basic of Programming - Level 2

Pattern continues

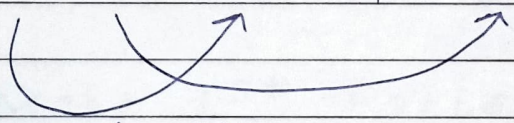
classmate Date Page 37

Full Pyramid

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

Rules

1. N = 5 (ROWS)
2. $r_0 \rightarrow$ 4 SP, 1 ★
   $r_1 \rightarrow$ 3 SP, 2 ★
   $r_2 \rightarrow$ 2 SP, 3 ★
   $r_3 \rightarrow$ 1 SP, 4 ★
   $r_4 \rightarrow$ 0 SP, 5 ★



n-row-1 row+1

Code:

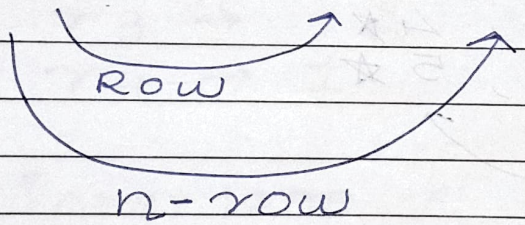
```
for(int row=0 ; row<n ; row++) {
    // space
    for(int col=0 ; col<n-row-1; col++ ){
        cout<< "_" ;
    }
    // stars
    for(int col=0;col<row+1 ; col=col+1) {
        cout << "*_" ;
    }
    cout<< endl;
}
return 0;
}
```

Inverted Pyramid

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

Rules

1. N = 4
2. r0 → 0 space, 4 ✱
   r1 → 1 space, 3 ✱
   r2 → 2 space, 2 ✱
   r3 → 3 space, 1 ✱

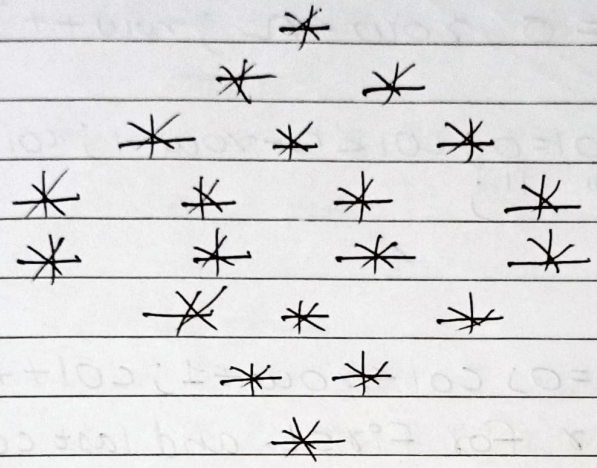


Code:

```
for(row =0; row<n; row++) {
    //spaces
  for(int col=0; col<row ; col++) {
     cout << " ";
  }

  //stars
  for( int col=0; col<n-row ; col++) {
     cout << " * ";
  }
  cout << endl;
}
```

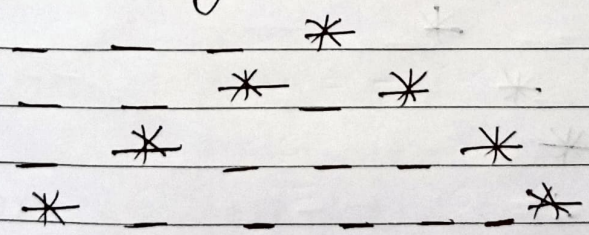
Diamond Pattern



Take num
suppose, num = 8
then our n = num/2
Hence, n = 4

Make 1st Pattern
then make inverted Pattern

Hollow Pyramid



Rules
1. N = 4
2. r0 → 3 sp, 1 ✱
   r1 → 2 sp, 1 ✱, 1 sp, 1 ✱
   r2 → 1 sp, 1 star, 3 sp, 1 ✱
   r3 → 1 star, 5 sp, 1 star

Code:

```
for(int row=0; row<n; row++){
    //space
    for(int col=0; col<n-row-1; col=col+1){
    cout<<" ";
    }
    //stars
    for(int col=0; col<row+1; col++){
    // Print star for first and last col
    if(col==0 || col==row+1-1){
    cout<<"* ";
    }
    else{ // For every col between 1st & last
    cout<<" ";
    }
}
```

Inverted hollow Pyramid

```
* - - - - *
 _ * _ _ *
 _ _ * * _
 _ _ _ * _
```

Rules

(1) N = 4

(2) 1st col & last col is ✱

Baaki sab space

↓

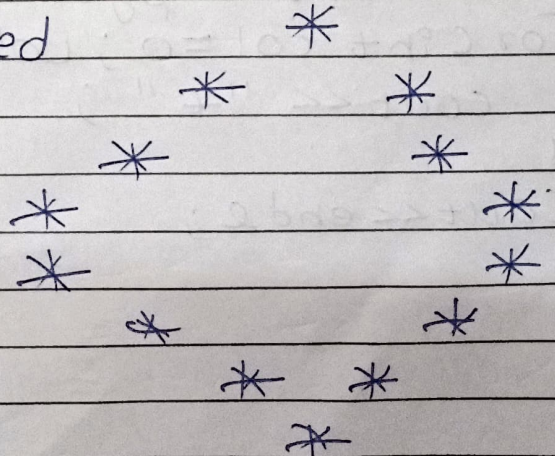
(in between)

Code:

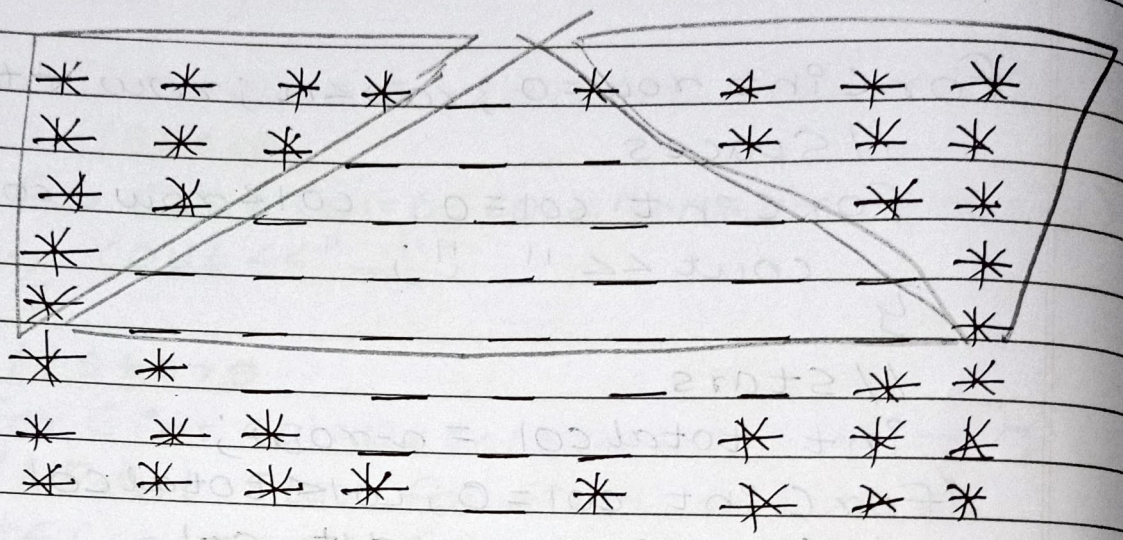
```
for(int row=0; row<n; row++){
    //spaces
    for(int col=0; col<row; col++){
        cout<<" ";
    }
    //stars
    int totalcol = n-row;
    for(int col=0; col<totalcol; col++){
        //if first or last col
        if(col==0 || col==totalcol-1){
            cout<<"* ";
        }
        else{
            cout<<" ";
        }
    }
    cout<<endl;
}
```

Hollow Diamond

Normal + Inverted



Flipped solid Diamond



Code:

```
for (int row=0; row<n; row++) {
    //inverted pyramid 1
    for (int col=0; col<n-row; col++) {
        cout << "* ";
    }
    //full pyramid 1
    for (int col=0; col<2*row+1; col++) {
        cout << " ";
    }
    //inverted pyramid 2
    for (int col=0; col<n-row; col++) {
        cout << "* ";
    }
    cout << endl;
}
```

```
for(int row=0; row<n; row++) {
    // inverted pyramid 1
    for(int col=0; col<row+1; col++) {
        cout << "*";
    }
    for(int col=0; col<2*n-2*row-1;
                                col++) {
        cout << " ";
    }
    for(int col=0; col<row+1; col=col+1) {
        cout << "*";
    }
    cout << endl;
}
```

Fancy Pattern

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

(1) Row = 4

(2) r0 → 1 ch
r1 → 3 ch
r2 → 5 ch
r3 → 7 ch

→ row+1
→ row+1
Oh even

H.W → Down part of this fancy Pattern

Code:

```
for (int row=0; row<n; row++){
    //inner loop
  for (int col =0; col< 2*row+1 ; col++){
     if (col%2 ==1){
        //odd number
        cout<< "*";
    }
    else {
      cout << row+1;
    }
}
cout<< endl;
}
```

Alternate solution homework

Inverted hollow

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

code:

```
for (int row=0; row<n; row++) {
   for (int col=0; col<n-row; col++) {
     if (row== 0 || row ==n-1) {
       cout << "*";
     }
   else {
     if ((col==0 || col ==n-row-1) {
       cout << "*" ;
     }
     else {
       cout << " ";
     }
   }
}
cout<<endl;
}
```

Home work

```
1
1   2
1   _   3
1   _   _   4
1   2   3   4   5
```

N = 5
r_0 → 1 ch
r_1 → 3 ch
r_2 → 4 ch
r_3 → 5 ch
r_4 → 5 ch

Alphabet Pattern

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

Typecasting

```
int number = col+1
char ch = col+1 + 'A' - 1
          ~~~~~
          number
```

Now it is mapped with the Alphabets.

Code:

```
for (int row=0; row<n; row++) {
    char ch;
    for (int col=0; col<=row+1; col++) {
        int num = col+1;
        ch = num + 'A' - 1;
        cout << ch;
    }
    //jab tak A tak nhi pohocte
    //tab tak print karenge
    for (char alphabet = ch; alphabet > 'A';) {
        alphabet = alphabet - 1;
        cout << alphabet;
    }
    cout << endl;
}
```

Homeworks

Total 8 problems as Homework.