

Prime factorization

1440

2 2 2 2 2 3 3 5

	2	1	4	4	0
	2	7	2	0	
	2	3	6	0	
	2	1	8	0	
	2	9	0		
	3	4	5		
	3	1	5		
	5	5			
					1

2 2 2 2 2 3 3 5

fac
2
3
4
5

$n = 1440$

$n \bmod 6$

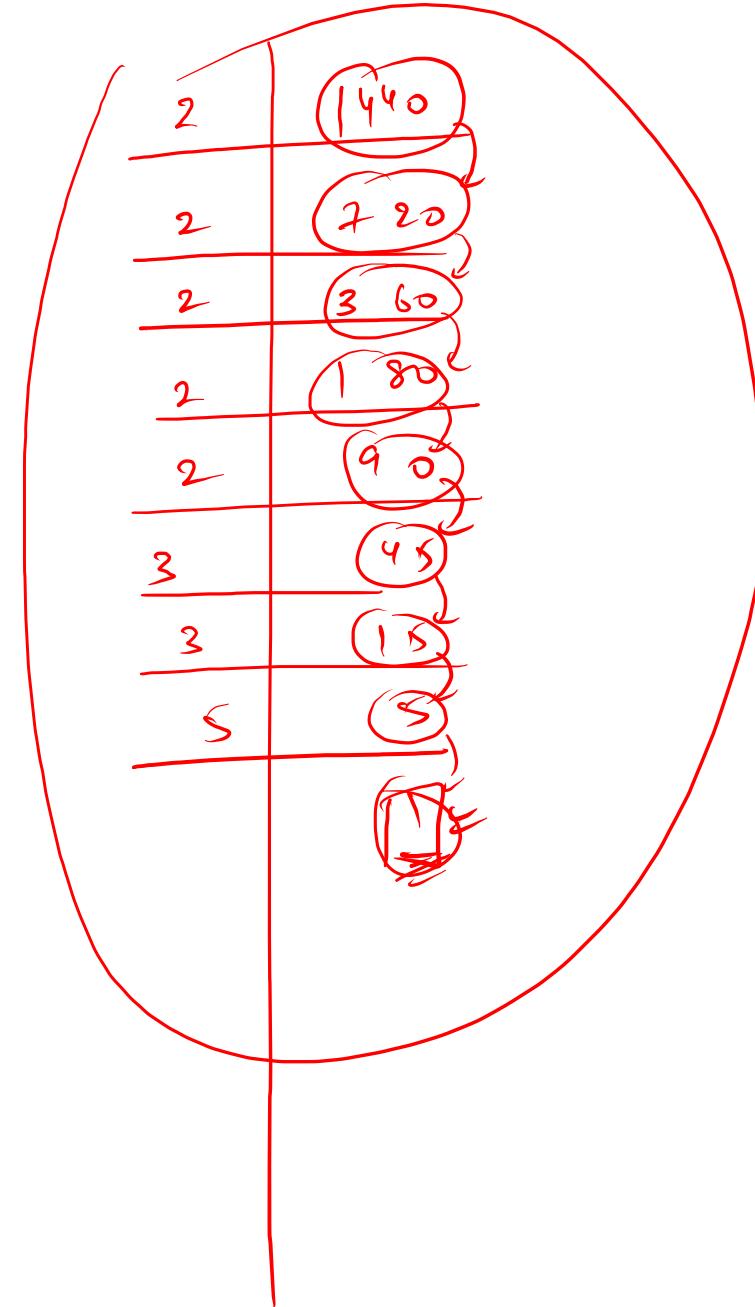
1440

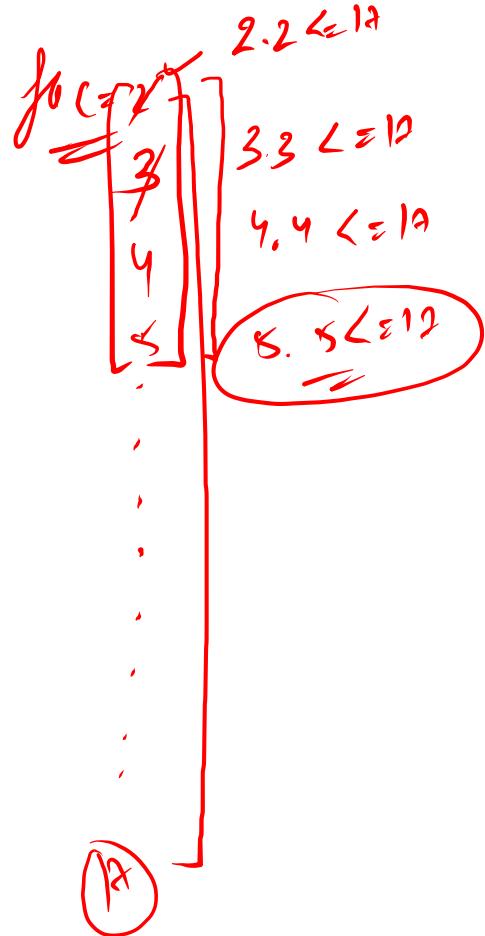
2 - n

```
for(int fac = 2 ; fac <= n ; fac++){
    while(n%fac==0){
        n = n/fac;
    }
    System.out.print(fac + " ");
}
```

$$\frac{P \cdot q = n}{\sqrt{P} > \sqrt{n}}$$

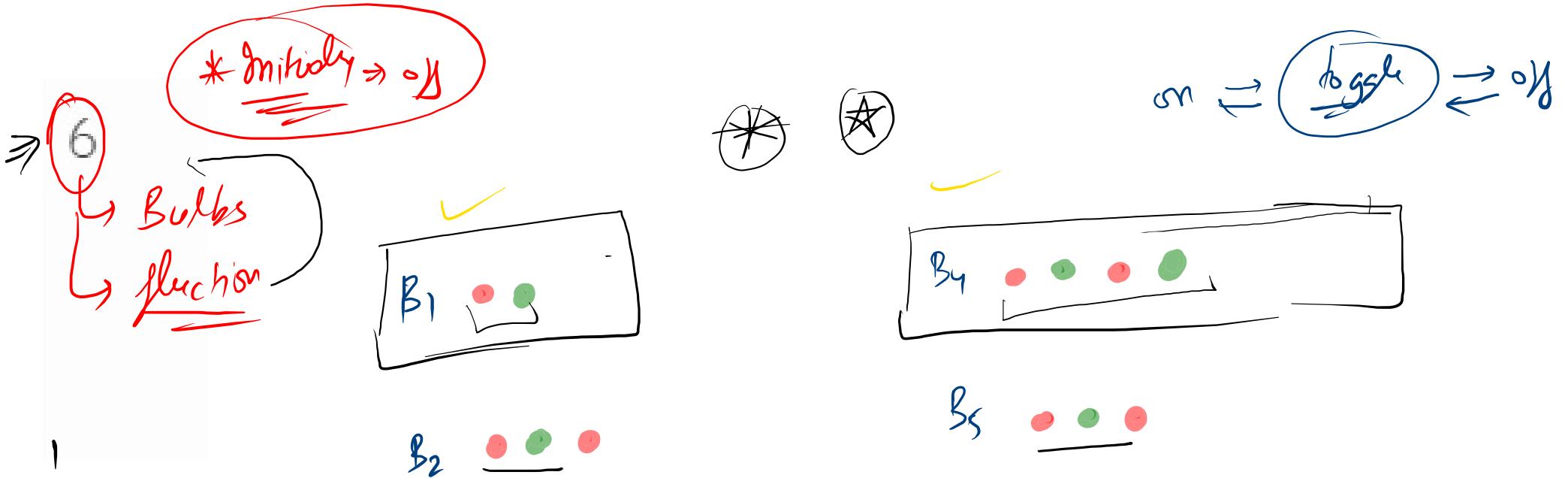
$\sqrt{q} > \sqrt{n}$





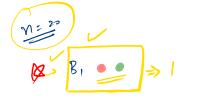
$$\begin{array}{r} 2 \\ \hline 34 \\ 12 \end{array}$$

2



toggle even \rightarrow on static

toggle odd \rightarrow off static



$\checkmark B_2 \rightarrow 2$

$\checkmark B_3 \rightarrow 2$

$\checkmark B_4 \rightarrow 3$

$B_5 \rightarrow 2$

$B_6 \rightarrow 4$

$B_7 \rightarrow 2$

$B_8 \rightarrow 4$

$B_9 \rightarrow 2$

$B_{10} \rightarrow 3$

$B_{11} \rightarrow 4$

$B_{12} \rightarrow 2$

$B_{13} \rightarrow 2$

$B_{14} \rightarrow 4$

$B_{15} \rightarrow 4$

$B_{16} \rightarrow 2$

$B_6 \rightarrow 4$

$B_{12} \rightarrow 2$

$B_{18} \rightarrow 4$

$B_{14} \rightarrow 2$

$B_{10} \rightarrow 3$

$B_{16} \rightarrow 2$

$B_{18} \rightarrow 4$

$B_{14} \rightarrow 2$

$B_{10} \rightarrow 4$

$B_{16} \rightarrow 4$

$B_{18} \rightarrow 4$

$\frac{\text{toggle}}{\text{multiple}}$

(why?)

$$\begin{array}{c|c} \sqrt{n} & \\ \hline 1 \times 36 & 36 \times 1 \\ 2 \times 18 & 18 \times 2 \\ 3 \times 12 & 12 \times 3 \\ 4 \times 9 & 9 \times 4 \end{array}$$

$$\begin{array}{c|c} \sqrt{n} & \\ \hline 6 \times 6 & \end{array}$$

$\sqrt{36} \Rightarrow 6$

odd toggles
odd factors
ON

Perfect square \Rightarrow factors $\Rightarrow 2 \cdot n + 1$

1 extra factor
with no duplicates

$$\begin{array}{c|c} \sqrt{n} & \\ \hline 1 \times 18 & 18 \times 1 \\ 2 \times 9 & 9 \times 2 \\ 3 \times 6 & 6 \times 3 \end{array}$$

$\sqrt{18} \approx 4$

Even toggle
even factors
OFF

Non perfect \Rightarrow factors $\Rightarrow 2 \cdot n$

factors \rightarrow toggle

$k! \Rightarrow n$

$1 \rightarrow k$
if (count \Rightarrow odd) {
 print(k);
}

2



$$\frac{2 \leq 1^2}{5, 5 \leq 1^2}$$

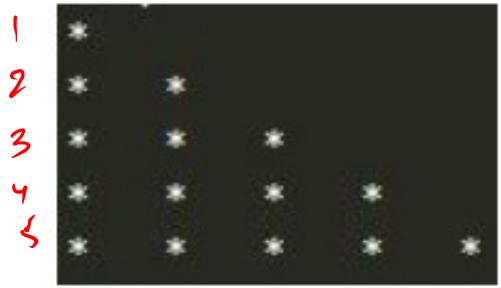
```
for(int fac = 2 ; fac*fac <= n ; fac++){
    while(n%fac==0){
        n = n/fac;
        System.out.print(fac+" ");
    }
}
```

$$\underline{\underline{n = 34}}$$

for =
By 5

$$\begin{array}{r} 34 \\ 2 \bigg| \\ 14 \end{array}$$

$n=5$



\overline{row}

T

2

3

4

5

\overline{nst}

T

2

3

4

5

$nst \rightarrow i$
 $\overline{row} \neq l$

n=5

2 nst > i → *

 | | x2 → * *

2 2 x2x3 →

3 3 x2x3x4

```
for(int r = 1 ; r <= n ; r++){
    int nst = r;
    for(int i = 1 ; i <= nst ; i++){
        System.out.print("*\t");
    }
    System.out.println();
}
```



" " \t "

row

1
2
3
4
5

nst

5
4
3
2
1

row + nst (n+1)

6
6
6
6
6

```
for(int r = 1 ; r <= n ; r++){
    int nst = n+1-r;
    for(int i = 1 ; i <= nst ; i++){
        System.out.print("*\t");
    }
    System.out.println();
}
```

row

new
newline

$$\text{row} + nst = n+1$$

$$nst = (n+1) - \text{row}$$

row = i

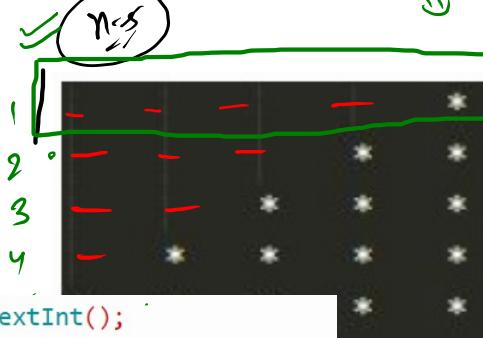
nst = ?

n = 5



Row

Space + stars



```

int n = scn.nextInt();

for(int r = 1 ; r <= n ; r++){
    int nsp = n-r , nst = r;

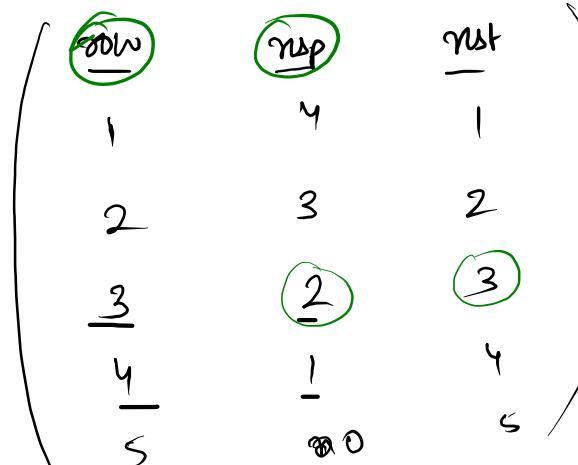
    for(int i = 1; i <= nsp ; i++){
        System.out.print("\t");
    }

    for(int i = 1; i <= nst ; i++){
        System.out.print("*\t");
    }

    System.out.println();
}

```

1, 2, 3, ..., nsp



row + nsp

nsp + nst = n

~~nsp = n - row~~

$r: 1 \rightarrow n$
 nst, nsp
 [Space, It]
 [nst, It]
 [[System.out.println()]]

5

row: spaces \rightarrow stars

M.W. Similar to pattern 3

Sample Output

1	*	*	*	*	*
2	-	*	*	*	*
3	-	-	*	*	*
4	-	-	-	*	*
5	-	-	-	-	*

row

nsp

nst

1	0	5
2	1	4
3	2	3
4	3	2
5	4	1

$$nsp = row - 1$$

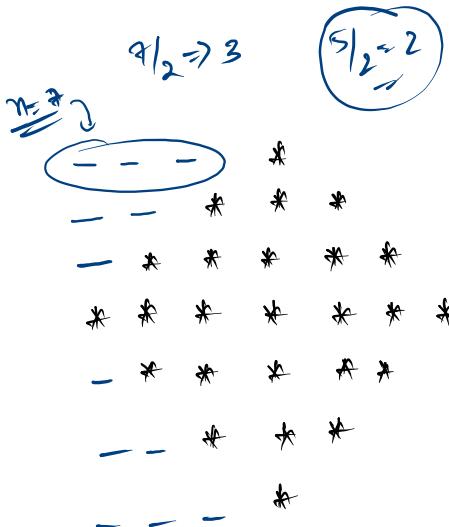
$$nst = nti - row$$

cols: 1 to n

nsp nst

{
 spaces
}

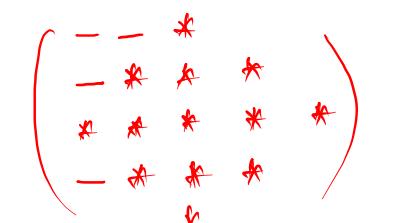
{
 stars
 println()



```

int row = 1, nst = 1, nsp = n/2;
while(row <= n){
    // code for each row
    for(int i = 1; i <= nsp; i++){
        System.out.print("\t");
    }
    for(int i = 1; i <= nst; i++){
        System.out.print("*\t");
    }
    // move to next row
    System.out.println();
    // preparation for next row
    row++;
}

```



$$\begin{cases} nsp = nsp - 1; \\ nst = nst + 2; \end{cases}$$

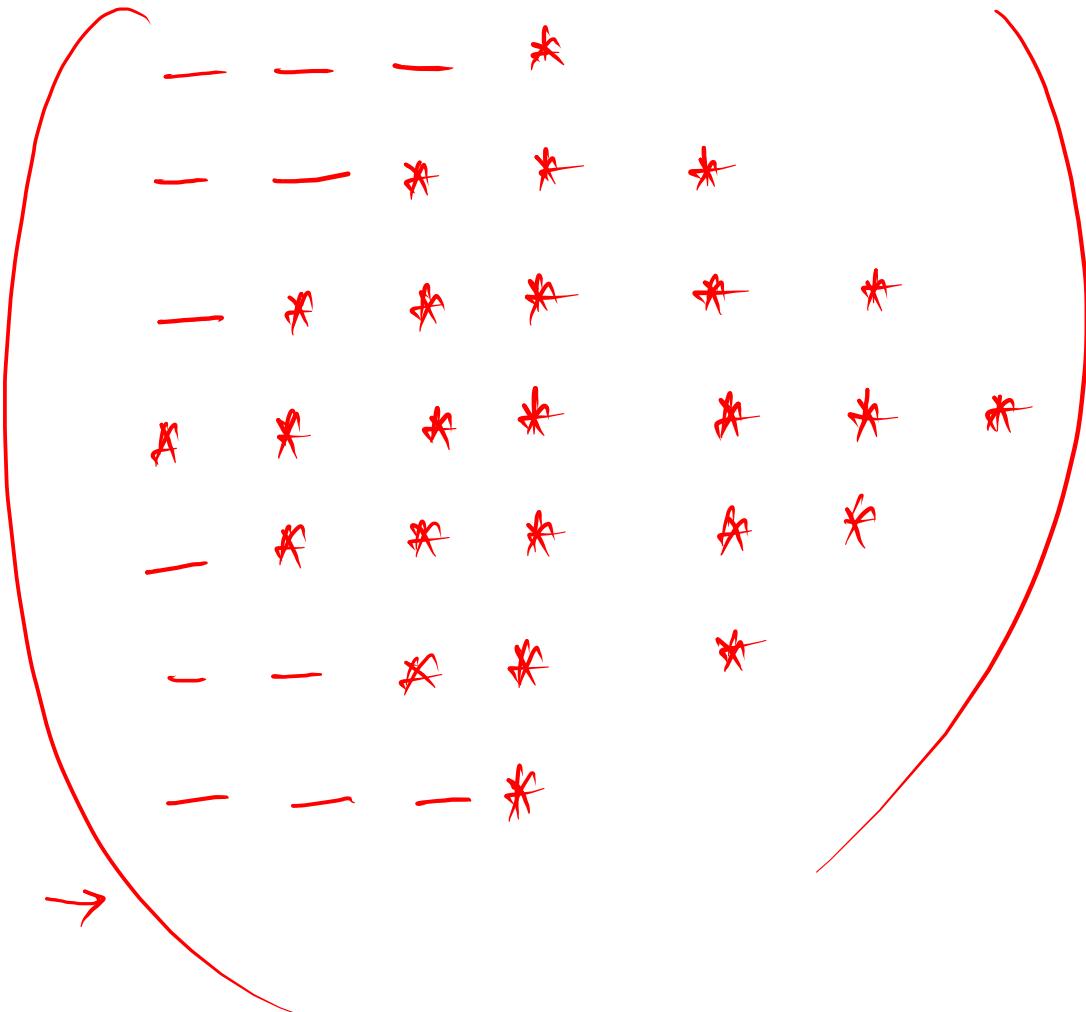
$$\begin{bmatrix} nsp = nsp + 1; \\ nst = nst - 2; \end{bmatrix}$$

$n \leq 2$
 if ($nrow \leq n/2$) {
 $nsp = nsp - 1;$
 $nst = nst + 2;$
 } else {
 $nsp = nsp + 1;$
 $\rightarrow nst = nst - 2;$
}

$$n=2$$

$$n/2 = 3$$

$$\text{down} = 348k \quad \text{out} = 878k \quad \text{up} = 10 \\ \begin{matrix} & 348 \\ & \times 8 \\ \hline & 2784 \end{matrix} \quad \begin{matrix} & 878 \\ & \times 10 \\ \hline & 8780 \end{matrix}$$



```
int n = scn.nextInt(); // q

int row = 1, nst = 1, nsp = n/2;

while(row <= n){

    // code for each row

    for(int i = 1 ; i <= nsp ; i++){
        System.out.print("\t");
    }
    for(int i = 1 ; i <= nst ; i++){
        System.out.print("*\t");
    }

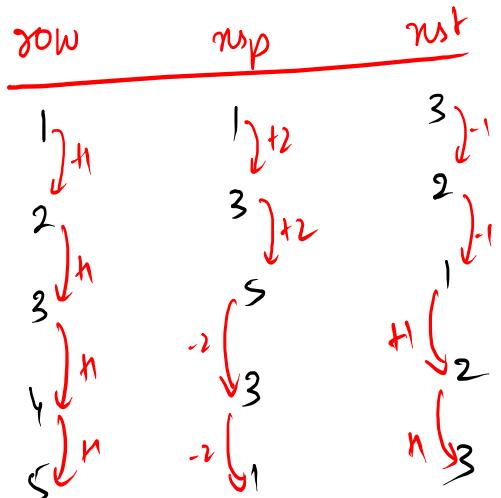
    // move to next row
    System.out.println();

    // preparation for next row
    if(row <= n/2){
        nsp = nsp - 1;
        nst = nst + 2;
    }else{
        nsp = nsp + 1;
        nst = nst - 2;
    }
}

row++;
}
```


row → stars

spans stars



5

Sample Output

```
* * * - * * *
* * - - - * *
* - - - - *
* * - - - * *
* * * - * * *
```

```
int row=1, nsp=1, nst =  $\frac{n}{2} + 1$ ;
while (row <= n) {
```

$\left[\begin{matrix} \text{stars} \\ \text{spaces} \\ \text{stars} \end{matrix} \right]_{\text{row}}$ $\left[\begin{matrix} \text{stars} \\ \text{spaces} \\ \text{stars} \end{matrix} \right]_{\text{nsp}}$ $\left[\begin{matrix} \text{stars} \\ \text{spaces} \\ \text{stars} \end{matrix} \right]_{\text{nst}}$
Code for each row

System.out.println()

$\left[\begin{matrix} \text{if} (row <= n/2) \{ \\ \quad nsp = nsp + 2; \\ \quad nst = nst - 1; \\ \} else \{ \\ \quad nsp = nsp - 2; \\ \quad nst = nst + 1; \\ \} \\ \text{row}++;$
Preparation for next row

$\frac{n}{2}$	row	nsp	nst
1	1	1	3
2	2	3	2
3	3	5	1
4	4	3	2
5	5	1	3
6	6	4	1

.
* * * - * * *
* * - - - * *
* - - - - *
* * - - - * *
* * * - * * *

Invert a number

Hint ↵

$$\text{digit} = 6 \quad \text{pos} = 4$$

(28346751)

$$\begin{array}{r} (\text{digit}) \\ \text{num} = 2 8 3 4 \boxed{6} 7 5 1 \\ \hline \text{pos} = 8 7 6 5 \boxed{4} 3 2 1 \end{array}$$

$$= 2 \cdot 10^7 + 8 \cdot 10^6 + 3 \cdot 10^5 + 4 \cdot 10^4 + 6 \cdot 10^3 + 7 \cdot 10^2 + 5 \cdot 10^1 + 1 \cdot 10^0$$

Sample Output

(73425681)

$$\begin{array}{r} (\text{digit}) \\ \text{num} = 7 3 4 2 5 6 8 1 \\ \hline \text{pos} = 8 7 6 5 4 3 2 1 \end{array}$$

$$\Rightarrow 2 \cdot 10^7 + 3 \cdot 10^6 + 4 \cdot 10^5 + 2 \cdot 10^4 + 5 \cdot 10^3 + 6 \cdot 10^2 + 8 \cdot 10^1 + 1 \cdot 10^0$$

$$(\text{digit}, \text{pos}) \Rightarrow (\text{digit} \times 10^{\text{pos}-1}) \Rightarrow \left(\frac{\text{pos} \times 10^{\text{digit}-1}}{\text{Output}} \right)$$

$$4 \times 10^{6-1} \Rightarrow 4 \cdot 10^5$$

H.W.

pattern 4, 6

next class

wednesday @ 02:00 pm