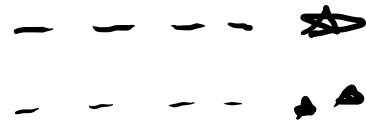
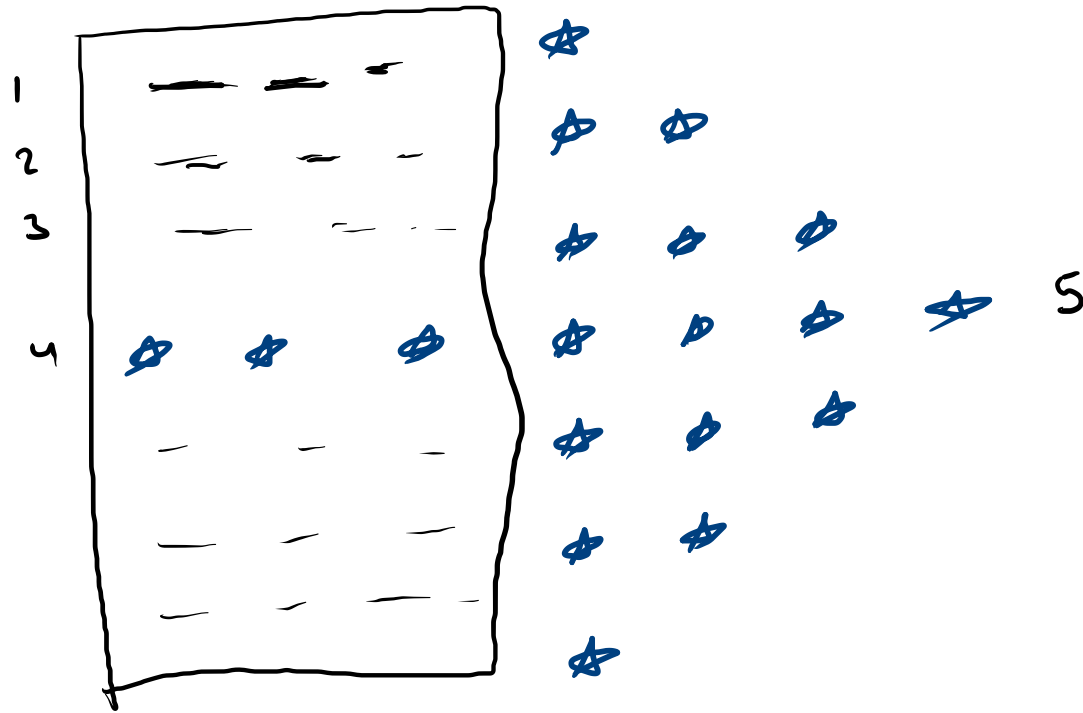
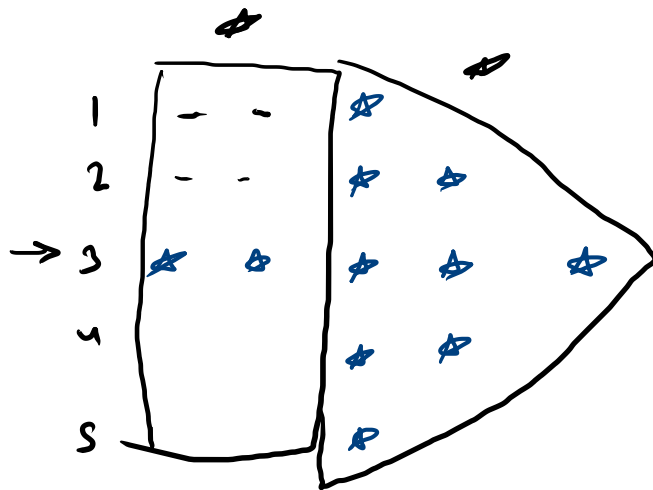


$$h = 5$$

$$n = 7$$

$$g$$



$$Space = h/2$$

$$sh = 2$$

$$\frac{7}{2} = 3$$

$$row = \frac{h}{2} + 1$$

$$\frac{9}{2} = 4$$

$$n = 2$$

$$i \in 1 \dots 2$$

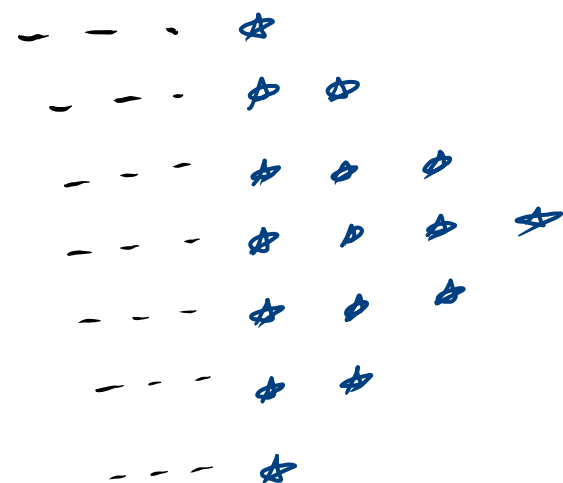


star

space

1 } +1
 2 } +1
 3 } +1
 4 } -1
 3 } -1
 2
 1

4/2 3



```

public static void main(String[] args) {
    Scanner scn = new Scanner(System.in);

    int n = scn.nextInt();

    int star = 1;
    int space = n/2;
    for(int i=1; i<=n; i++){
        for(int j=1; j<=space; j++){
            if(i == n/2+1){  $4 = 3$ 
                System.out.print("*\t");
            }else{
                System.out.print("\t");
            }
        }

        for(int j=1; j<=star; j++){
            System.out.print("*\t");
        }
        System.out.println();
        if(i<=n/2){  $3 <= 2$ 
            star++;
        }else{
            star--;
        }
    }
}

```

$$n = 5$$

$$\text{star} = 1 \ 2 \ 3 \ 2 \ 1$$

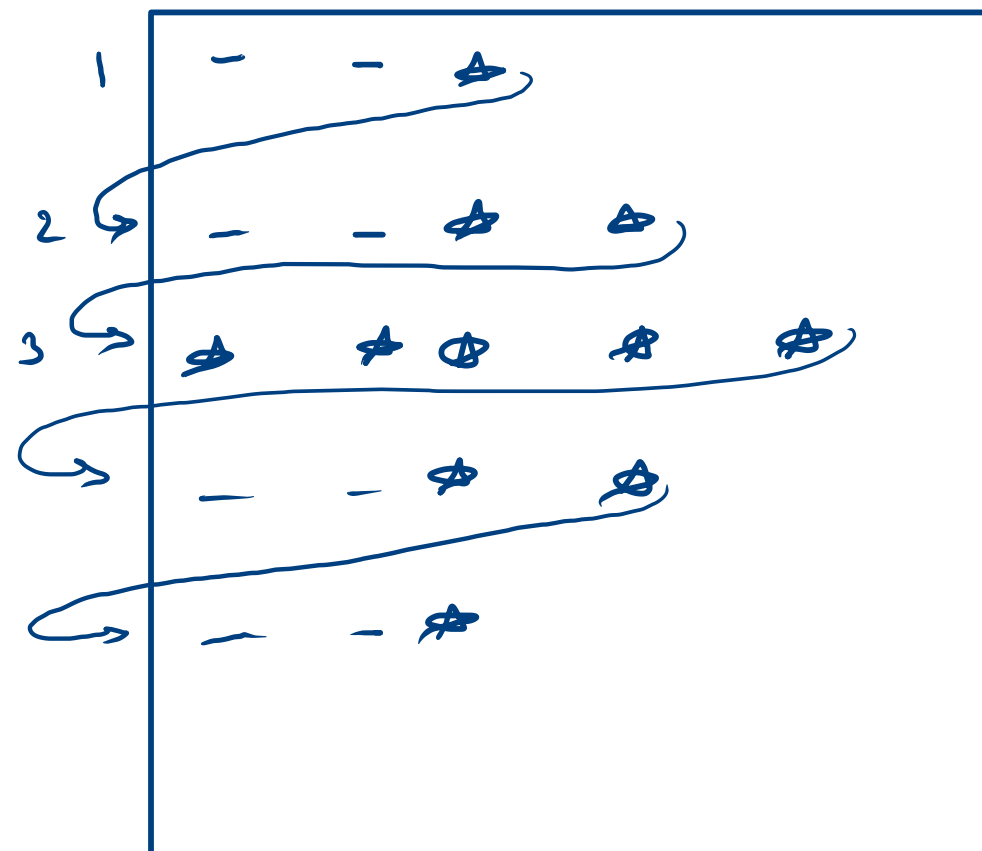
$$\text{space} = 2$$

$$i = 1 \ 2 \ 3 \ 4$$

$$4/2 = 2$$

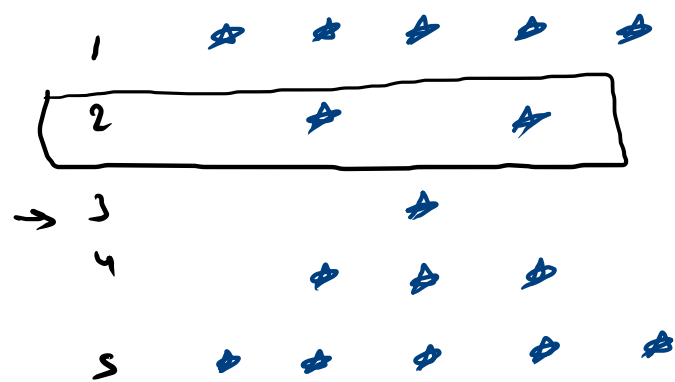
$$5/2 = 2$$

$$2 \times 1$$

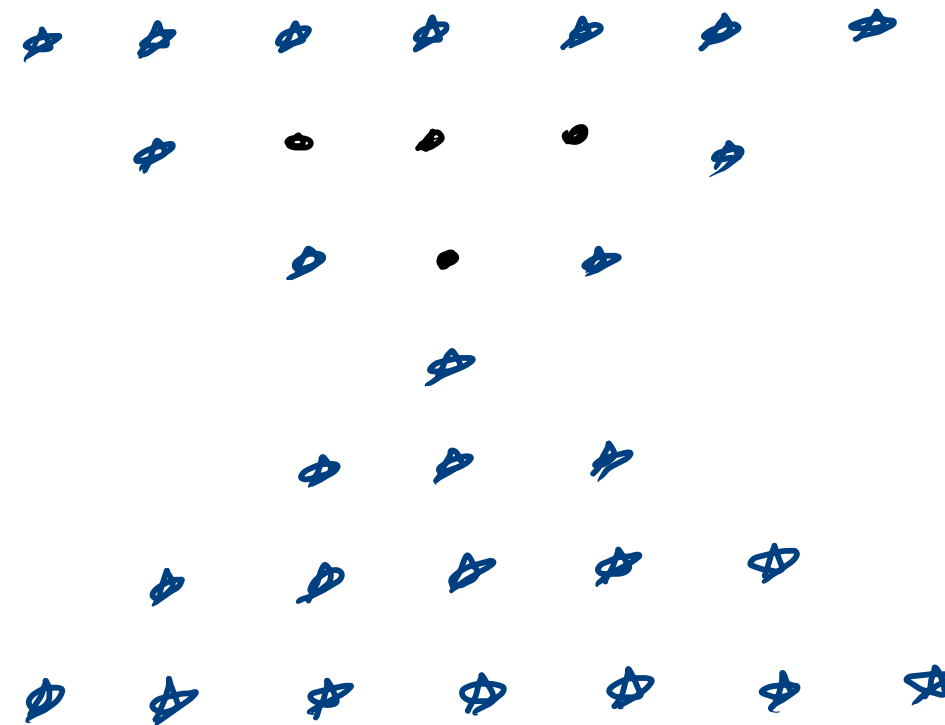


$$h = 5$$

$$(h/2 + 1)$$

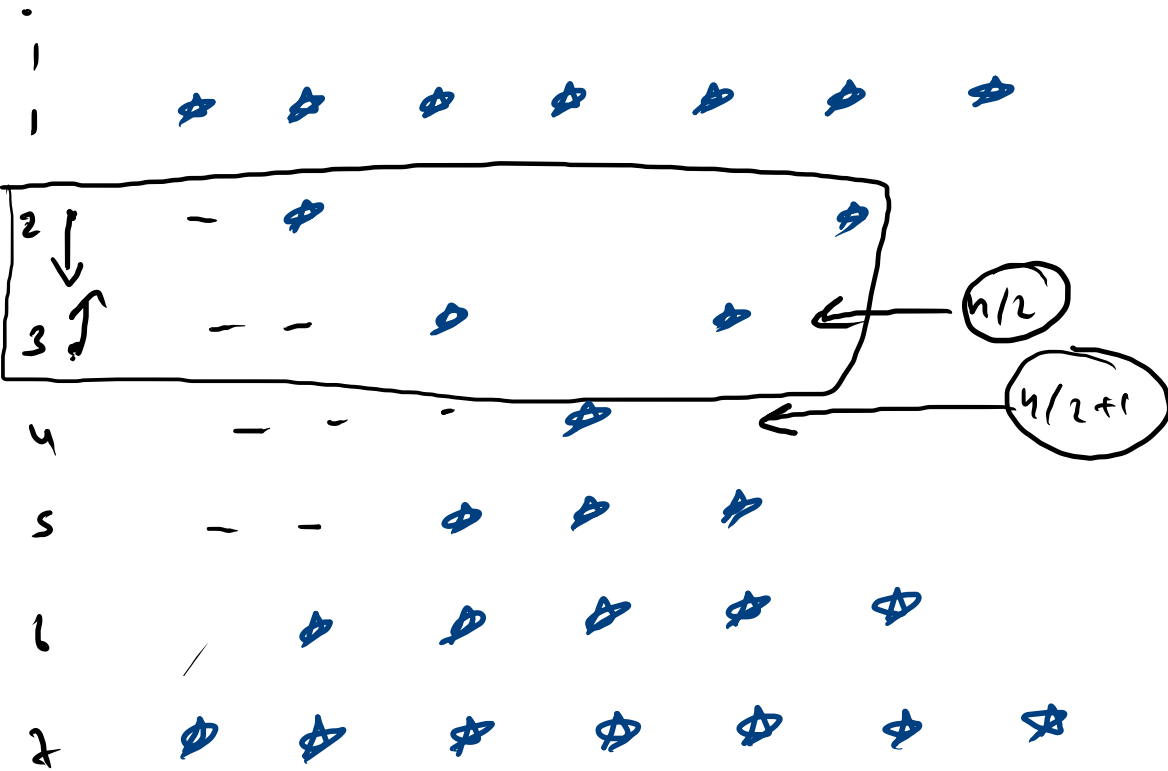


$$h = 2$$



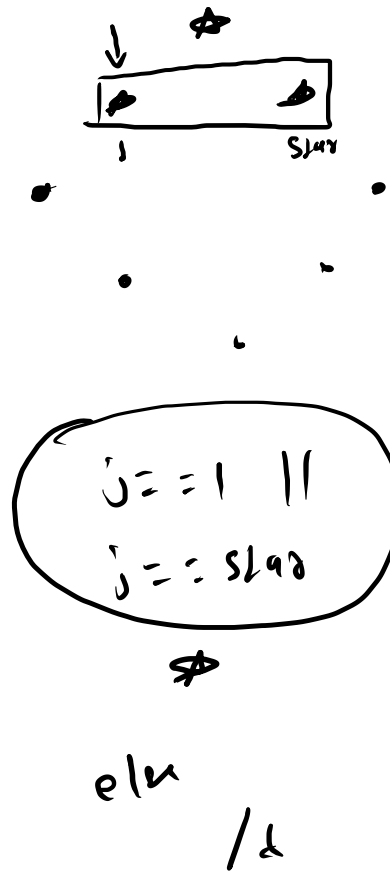
$h=2$

$2 \dots n/2$



star
 2 } -2
 5 } -2
 3 } -2
 1 } +2
 3 } +2
 5
 2

space
 0 } +1
 1 } +1
 2 } +1
 3 } -1
 2 } -1
 1 } -1
 0



```
if(i>=2 && i<=n/2){
```

```

int star = n;
int space = 0;

for(int i=1; i<=n; i++){
    for(int j=1; j<=space; j++){
        System.out.print("\t");
    }
    //
    for(int j=1; j<=star; j++){
        if(i>=2 && i<=n/2){
            if(j==1 || j==star){
                System.out.print("*\t");
            }else{
                System.out.print("\t");
            }
        }else{
            System.out.print("*\t");
        }
    }
    System.out.println();
    if(i<=n/2){
        space++;
        star = star-2;
    }else{
        space--;
        star = star+2;
    }
}

```

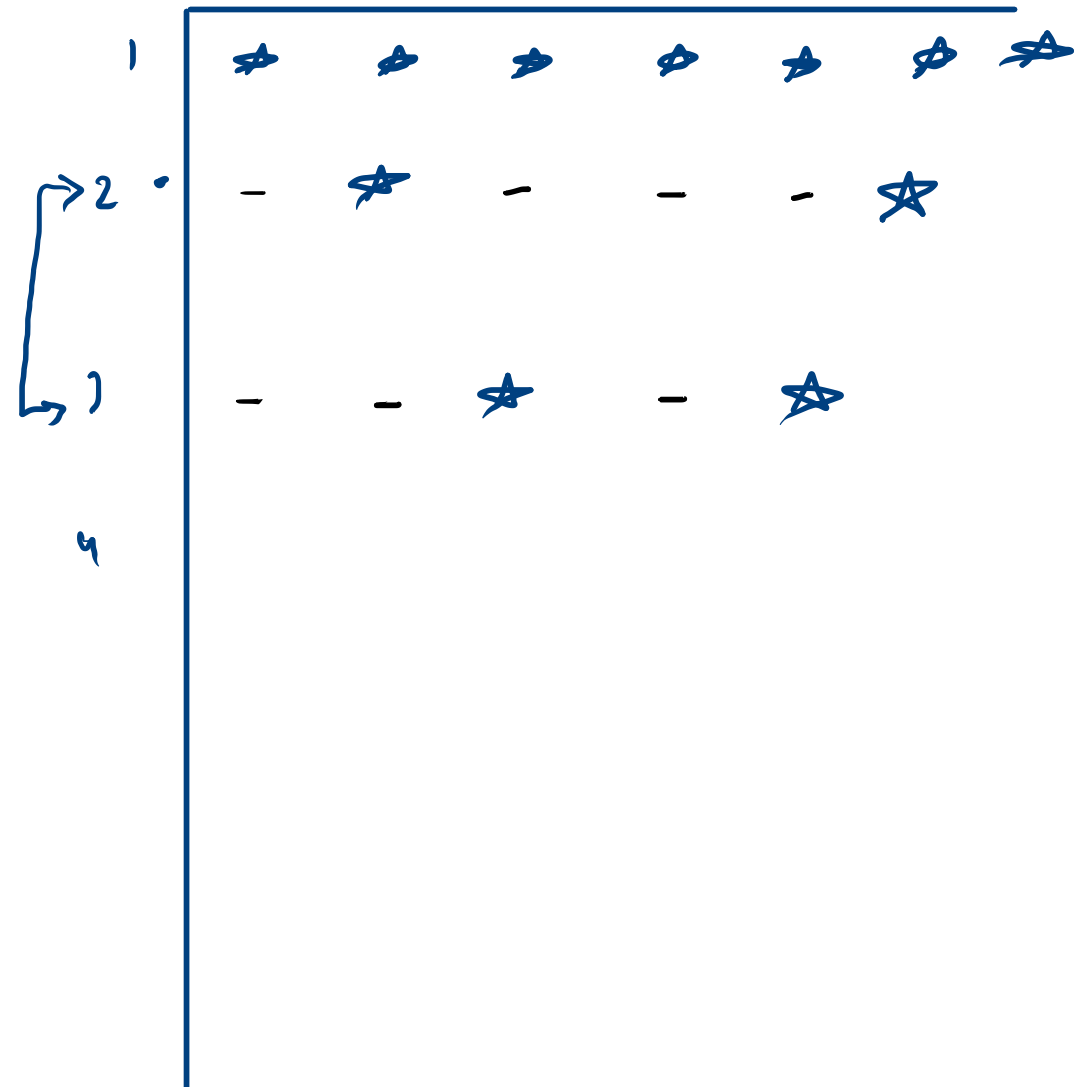
$n = 7$

$star = 7$

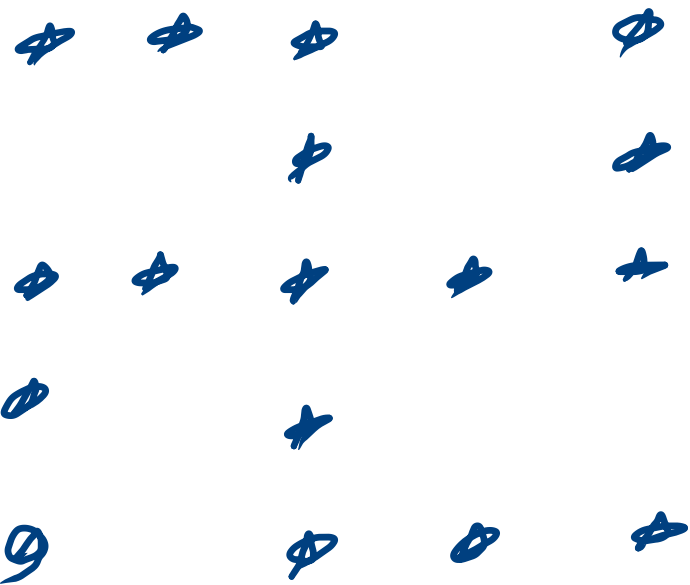
$space = 0$

$i = 1$

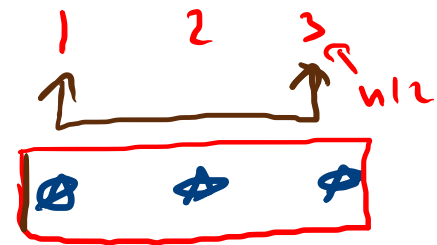
$j = 1$



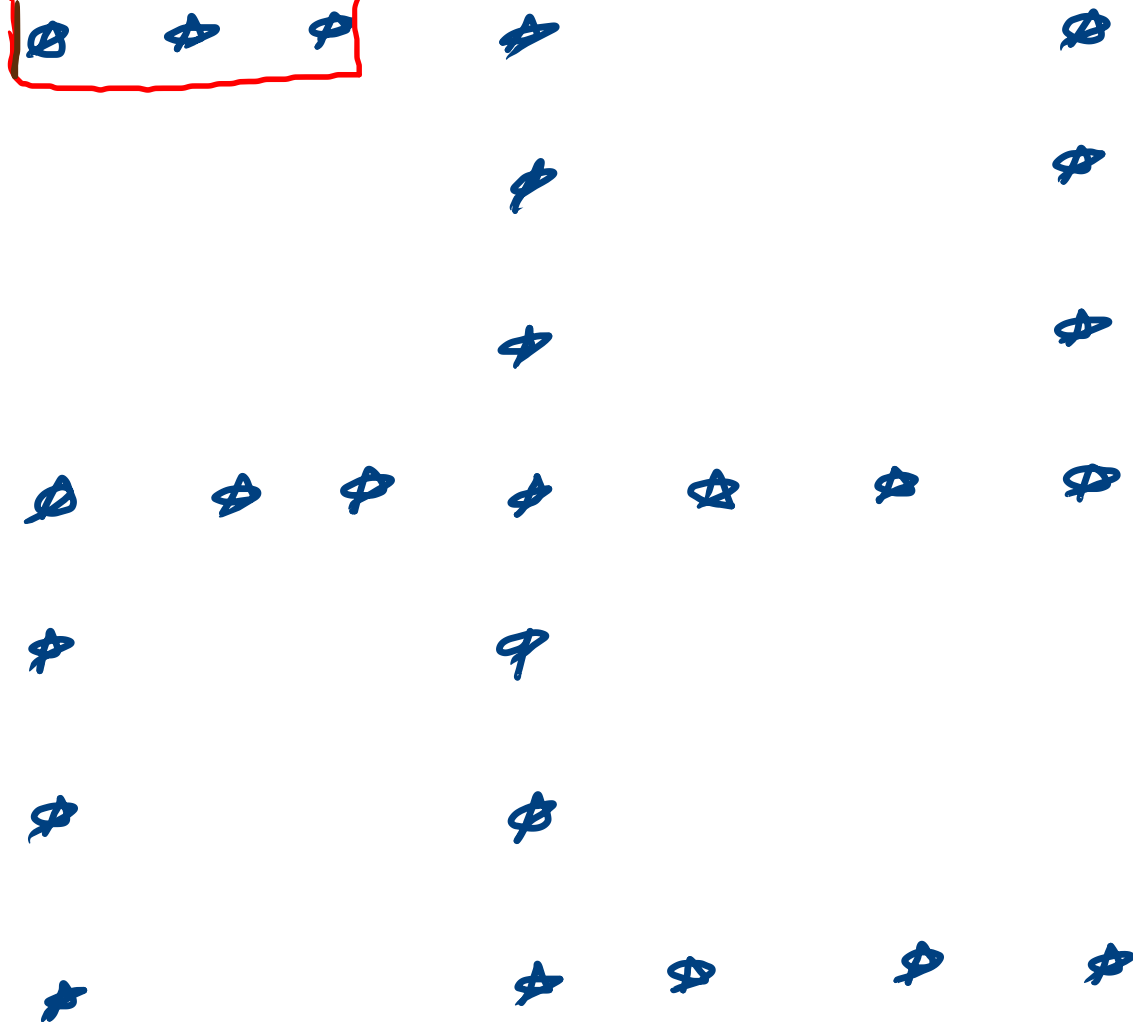
$h = 5$



$h = 2$



$n/2 + 1$



$$h = 5$$
$$h/2 + 1$$

↓

h/r

1 2

3

4

5



1

2

 $n^{n+1} \rightarrow$

3

4

5

☆ ☆

 $\leftarrow h/2 \pm 1$ 

$i = 1$

$$\beta\beta(j \geq 1 \quad \beta\beta j \leq n/2)$$
$$j = z \cdot h$$
$$88 \quad i \leq n/2$$

$$\begin{aligned} j &> 5 \\ j &\geq 6 \end{aligned}$$

☆ ☆



4

4


$$i = 2n \quad 8 \quad j > n/2$$

$\dot{u} = -1$ 88 $i > u/2$

Maths

int n
int i

Java

let $y = f(n)$
 $(24) = f(4)$

$$f(n) = n^2 + 4n + 3$$

$$f(3) = 3^2 + 4 \times 3 + 3$$

int fact (int n) {
 int ans = 1;
 for (int i = 1; i <= n; i++)
 ans = ans * i;
 return ans;
}

$$120 / (2 \times 6)$$

$$120 / 12$$

$$(10)$$

```
public static void main(String[] args) throws Exception {
    • int n = 5;
    ✓ int r = 2;
    ✓ int nf = fact(n);
    ✓ int rf = fact(r);
    ✓ int nmrf = fact(n-r);
    ✓ int nCr = nf / (rf * nmrf);
    ✓ System.out.println(nCr);
}
```

```
public static int fact(int x){
    int ans = 1;
    for(int i=1; i<=x; i++){
        ans = ans*i;
    }
    return ans;
}
```

main

10

Calc

~~nf = 6~~
~~n = 3~~

n = 10
nmrf = 6
rf = 2
nf = 120
r = 2
n = 5

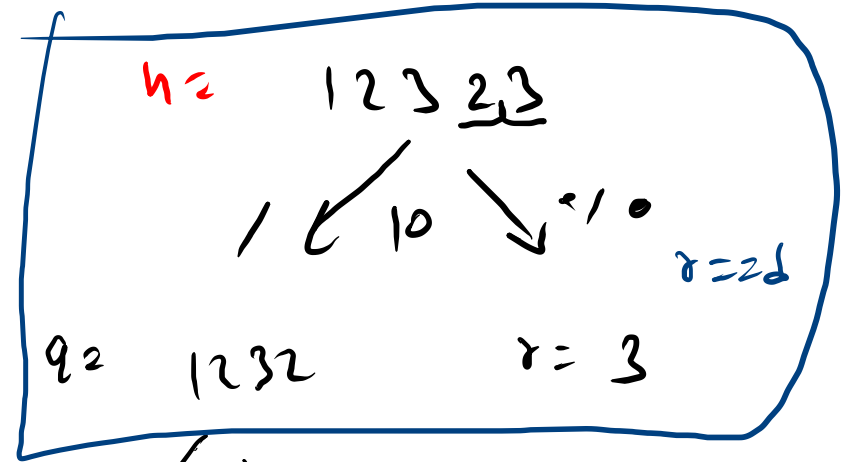
Stack

heap

h = 2 1 [✓] 3 2 8 [✓] 3 4 2 [✓] 3

d = 3

return 3



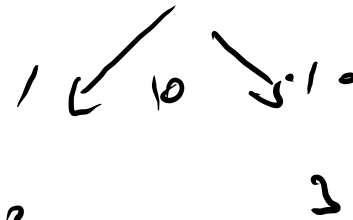
count = 0



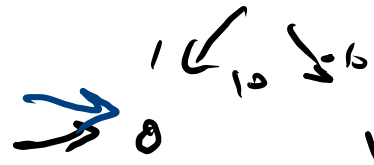
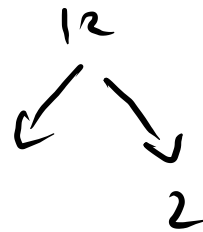
q = 1 2 3

r = 2

r = d



↙
r = 2d



1

2

2

2

Decimal

0 ... 9

Binary

0 1

hexadecimal

0 ... 9 A B C D E F

0 10
1 11
2 12
3
4
5
6
7
8
9

octal

0 ... 7

46

Decimal

0 --- 9

0	10 ⁰	20	30	40	100
1	11	21	31	41	
2	12		32	42	
3	13		33	43	
4	14		34	44	
5	15		35	45	
6	16		36	46	
07	17		37	47	
08	18		38	48	
09	19	29	39	49	

Binary

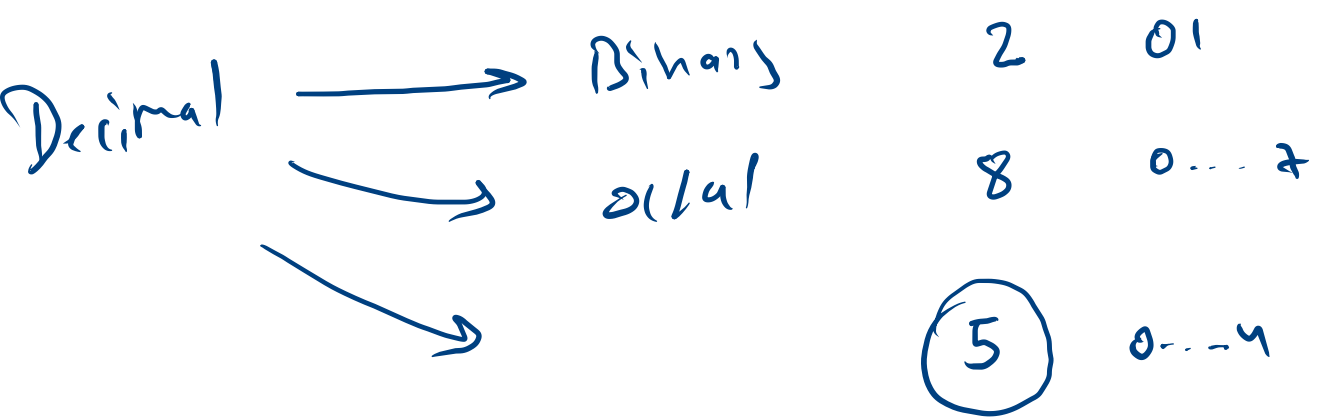
0 1

0	10	100	1000
1	011	101	
		110	
		0111	

Octal

0 --- 7

0	10	20	30	40	100
1	11				
2	12				
3					
4					
5					
6					
07	17	27			



$$(n)_{10} \rightarrow (?)_b$$

$$(57)_{10} \rightarrow (?)_2$$

$$n \rightarrow 57$$

$$b \rightarrow 2$$

$$(57)_{10} \rightarrow (111001)_2$$

2	57	
2	^q 28	$r_2 = 1$
2	^q 14	0
2	7	0
2	3	1
2	1	1
	0	1

111001

$$(n)_{10} \rightarrow ()_b$$

$$(123)_{10} \rightarrow ()_8$$

$$(123)_{10} \rightarrow (123)_8 =$$

$$\begin{array}{r} 8 \overline{) 123} \end{array}$$

$$\begin{array}{r} 123 \\ 10 \ 0 \rightarrow 1 \times 10^2 \\ 2 \ 0 \rightarrow 2 \times 10^1 \\ 3 \rightarrow 3 \times 10^0 \end{array}$$

ans

int pow = 0;
int ans = 0;
while

while (n != 0) {

$$q = n / b$$

$$r = n \% b$$

$$ans = ans * 10 + r$$

pow++;

n = q

}

b	123	
8	15	3
8	1	7
	0	1

$$+ 3 \times 10^0$$

$$+ 7 \times 10^1$$

$$+ 1 \times 10^2$$


```

public static void main(String[] args) {
    Scanner scn = new Scanner(System.in);
    ✓ int n = scn.nextInt();
    ✓ int b = scn.nextInt();
    ✓ int dn = getValueInBase(n, b)
    ✓ System.out.println(dn);
}

public static int getValueInBase(int n, int b){
    • int pow = 0;
    int ans = 0;

    while(n != 0){
        ✓ int q = n/b;
        ✓ int r = n%b;

        ✓ int multi = (int)Math.pow(10, pow);
        ✓ ans = ans + r*multi;

        pow++;
        n=q;
    }

    return ans;
}

```

$(23)_{10} \rightarrow (???)_b$

$n = 23$

$b = 4$

$pow = 0 + 2 + 3 + 4$

$ans = 0 + 2 + 0 + 2 + 1021$

$ans = 21 + 1000$
 $= 1021$

main

$dn = 1021$
 $b = 4$
 $n = 23$

4	23	
4	18	1
4	4	2
4	1	0
	0	1

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

1 2 3 4 5