

← functions →

```
main() {
```

```
    . . . . .
```

```
    while (a1 > 0) {
```

```
        | sum1 = a1 * 10
```

```
        | a1 = a1 / 10
```

```
        | 3
```

```
        Print (sum1);
```

```
    . . . . .
```

```
    while (a2 > 0) {
```

```
        | sum2 = a2 * 10
```

```
        | a2 = a2 / 10
```

```
        | 3
```

```
        Print (sum2);
```

```
    . . . . .
```

```
    while (a3 > 0) {
```

```
        | sum3 = a3 * 10
```

```
        | a3 = a3 / 10
```

```
        | 3
```

```
        Print (sum3);
```

```
    }
```

Problems

↳ Redundancy

↳ Readability,

↳ Maintainability.


```

public static int sum3(int a, int b){
    System.out.println(a+b);
    return 0;
}

public static void main(String[] args) {
    int a = 10;
    int b = 15;
    // int ans = int2Sum(a,b);
    // System.out.println(ans);
    // System.out.println(int2Sum(a,b));
    // System.out.println(sum3(a,b));
    sum3(a,b);
}

```

Handwritten notes: Red arrows point to 'a' and 'b' in the sum3 function signature. Red circles around '10' and '15' in the main function. Red circles around 'a' and 'b' in the sum3(a,b) call. Red text 'print : ?' below the call. Blue arrows point to the 'int' keyword in the variable declarations.

int x = 05; 5

Handwritten notes: Red circle around '05'. Red arrow pointing to '5' with label 'octal'.

int x = 08

int x = 017; 17

Handwritten notes: Red circle around '017'. Red arrow pointing to '17' with label 'decimal'. Red circle around '17' with label 'octal'.

row + st + 1 = m + 1
st + 1 = (m + 1) - row

m = 5

```

***** // 0 spaces
****_**** // 2 spaces
***___*** // 4 spaces
**_____** // 6 spaces
*_____* // 8 spaces

```

row	st+1	st	st+2
1	5	0	5
2	4	2	4
3	3	4	3
4	2	6	2
5	1	8	1

*Handwritten notes: Blue circle around the first row (1, 5). Red text 'm+1' above the first row. Red text '2*row - 2' below the table.*

2 * row - 2

n = 5

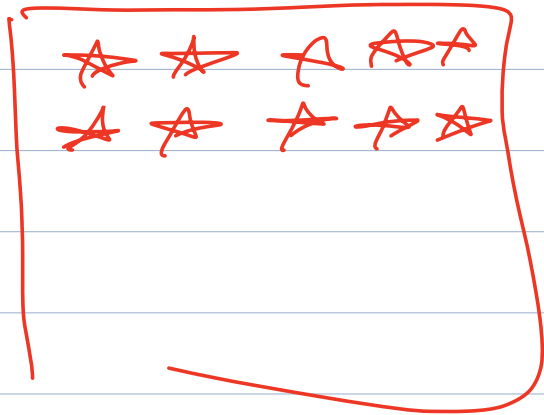
	0	0	0	0	5	0	0	0	0
1	0	0	0	0	4	8	12	0	0
2	0	0	3	6	9	12	15	0	0
3	0	2	4	6	8	10	12	14	0
4	1	2	3	4	5	6	7	8	9

- - - - -
 - - - - -
 - - - - -
 - - - - -
 - - - - -
 - - - - -

```

2 for (
    int col = 1;
    for (
        col = 24, inner = col;
        for (
            print (col);
            col += inner;
        for (
  
```

1	2	3	4	5
11	12	13	14	15
21	22	23	24	25
16	17	18	19	20
6	7	8	9	10



startno = 1;

for (i = 1; i <= n; i++) {

int colno = startno;

for (j = 1; j <= n; j++) {

print ('★');
colno++;

3
3

startno += 10;

startno
= 11

3

zero

1	1	2	3	4	5
2	11	12	13	14	15
3	21	22	23	24	25
<hr/>					
	16	17	18	19	20
	6	7	8	9	10

1 2 3 4 5 6

7 8 9 10 11 12

143

$$1^3 + 2^3 + 3^3 \Rightarrow \underline{143}.$$

no. of digits (n).

$$\underline{1}^3 +$$

$$\underline{\quad} \underline{\quad}$$

ischeck (n) {

x = no. of digits(n) → 3

while (n > 0) {

dig = n % 10;

sum += Math.pow(dig, x);

n = n / 10;

if (sum == n) {

return true;

get A:

for (i = 1; i <= n; i++)

if (ischeck(i))

print(i);

$$\begin{array}{r} 143 \rightarrow 3 \\ \downarrow \\ 1^3 + 4^3 + 3^3 = 1 \end{array}$$