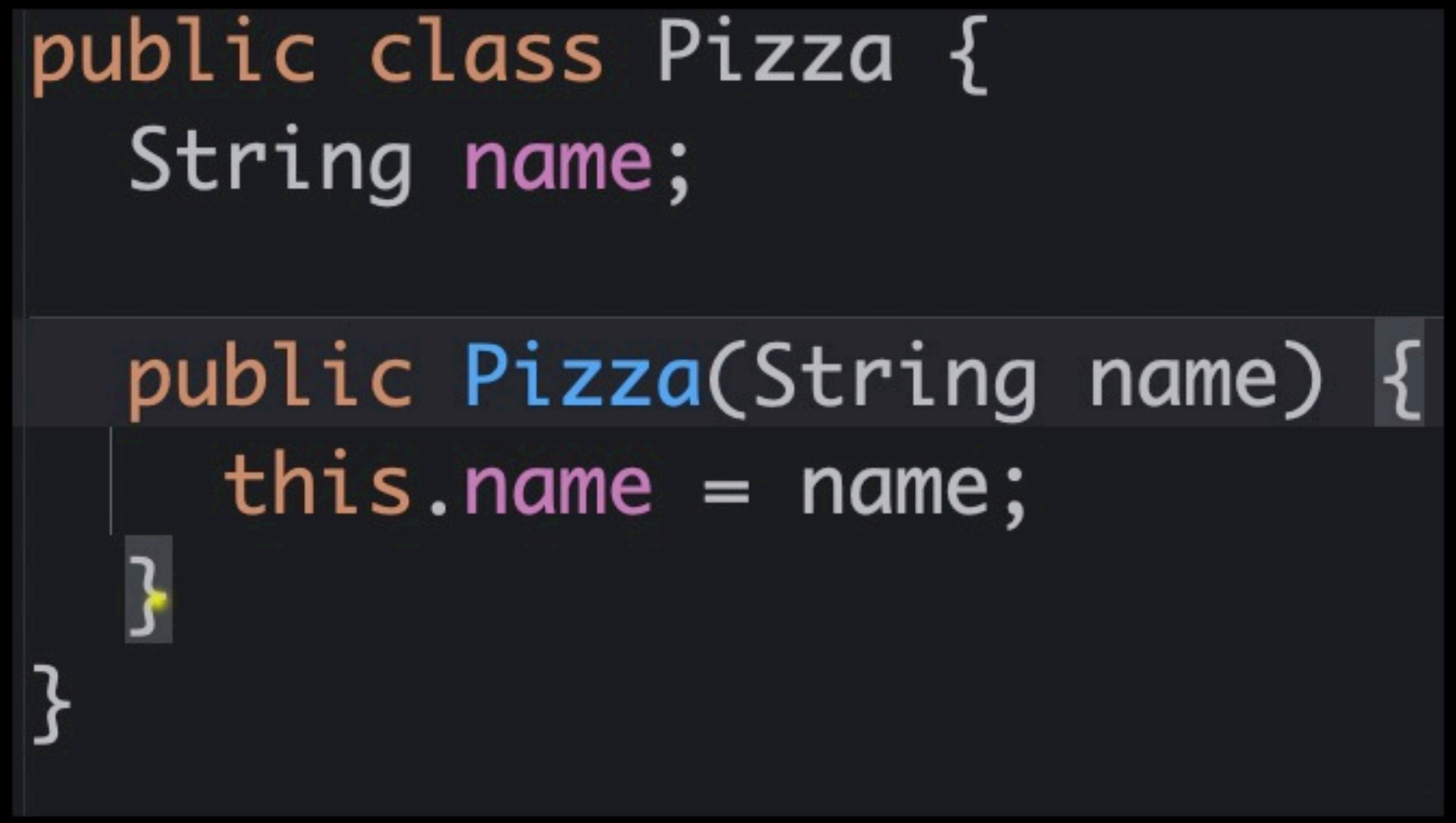
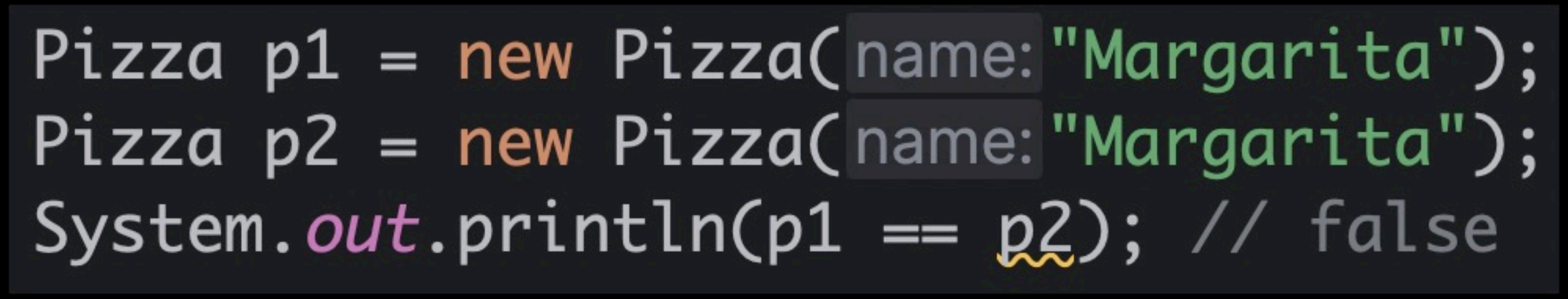


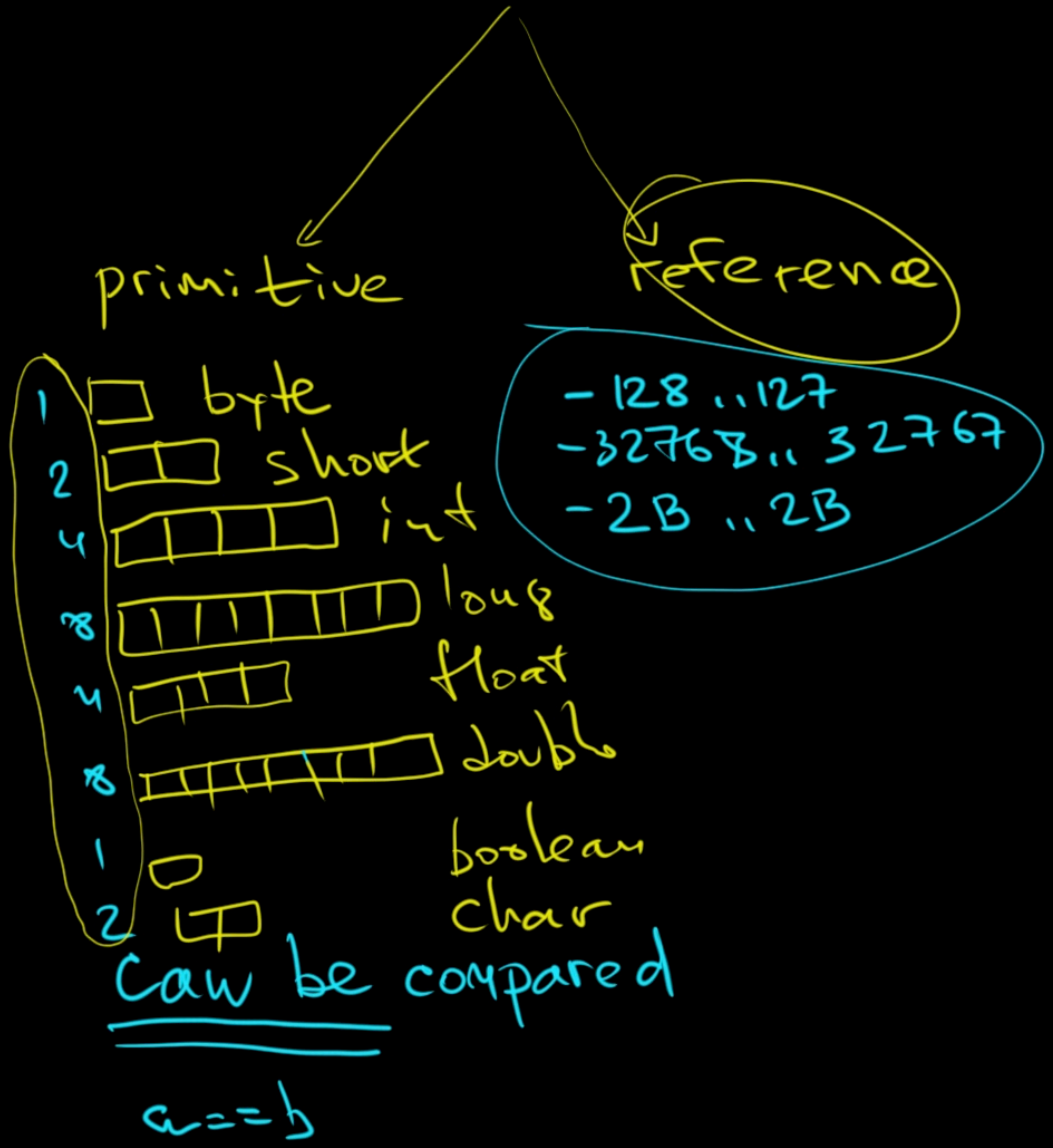
```java
public class Pizza {
  String name;

  public Pizza(String name) {
    this.name = name;
  }
}
```

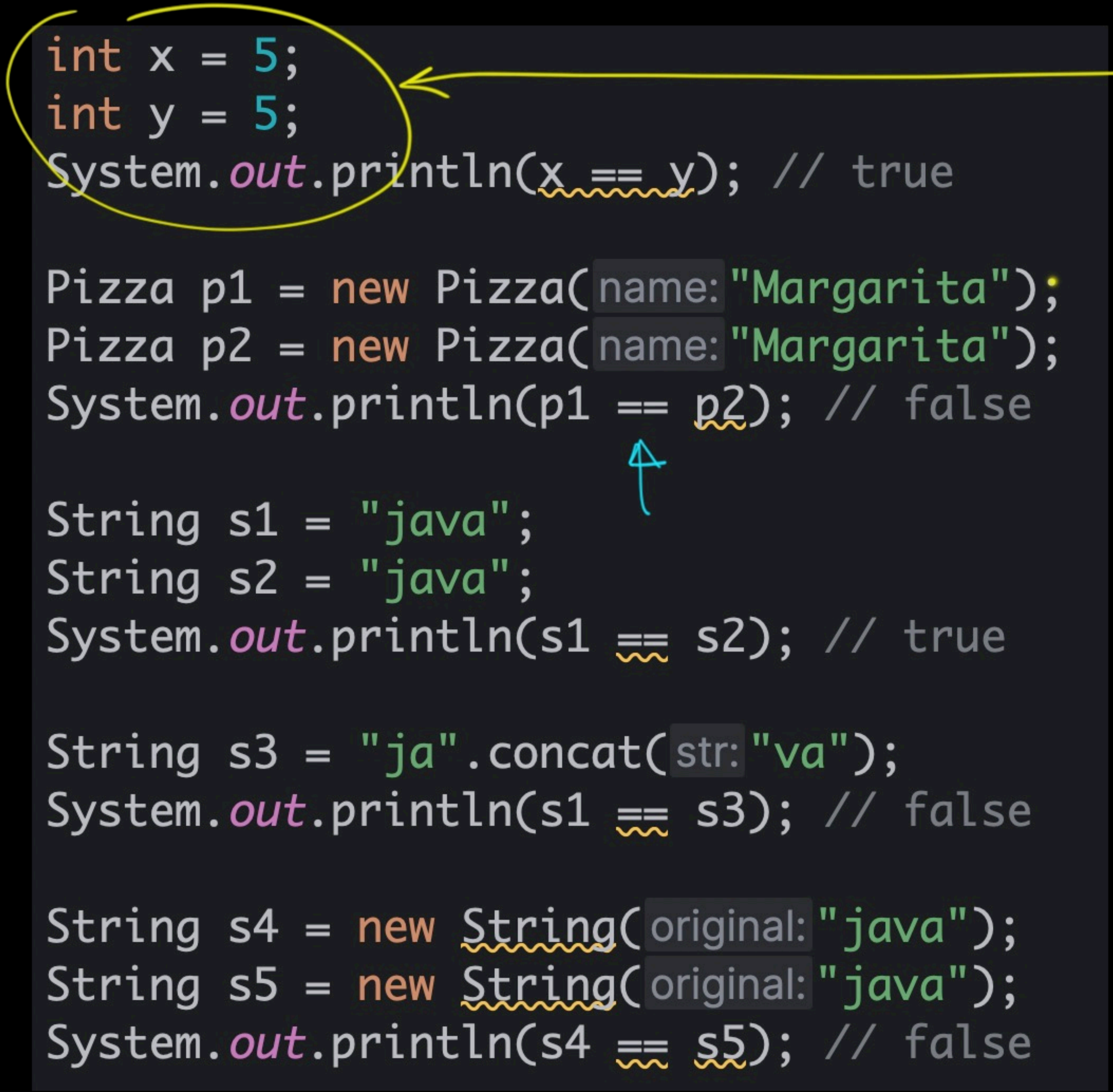


```java
Pizza p1 = new Pizza(name: "Margarita");
Pizza p2 = new Pizza(name: "Margarita");
System.out.println(p1 == p2); // false
```

1 2 4 8 16 32 64 128 256 512 1024 2048
5 8 10

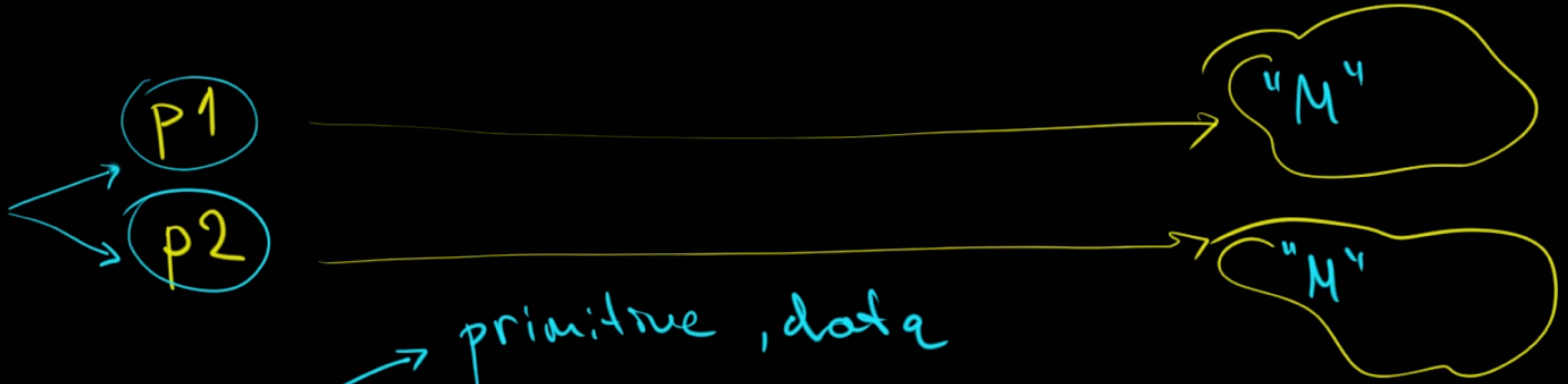
4096 8192 16384 32768 65535
16





```java
int x = 5;
int y = 5;
System.out.println(x == y); // true

Pizza p1 = new Pizza( name: "Margarita");
Pizza p2 = new Pizza( name: "Margarita");
System.out.println(p1 == p2); // false

String s1 = "java";
String s2 = "java";
System.out.println(s1 == s2); // true

String s3 = "ja".concat( str: "va");
System.out.println(s1 == s3); // false

String s4 = new String( original: "java");
String s5 = new String( original: "java");
System.out.println(s4 == s5); // false
```

primitive, fixed size ✓



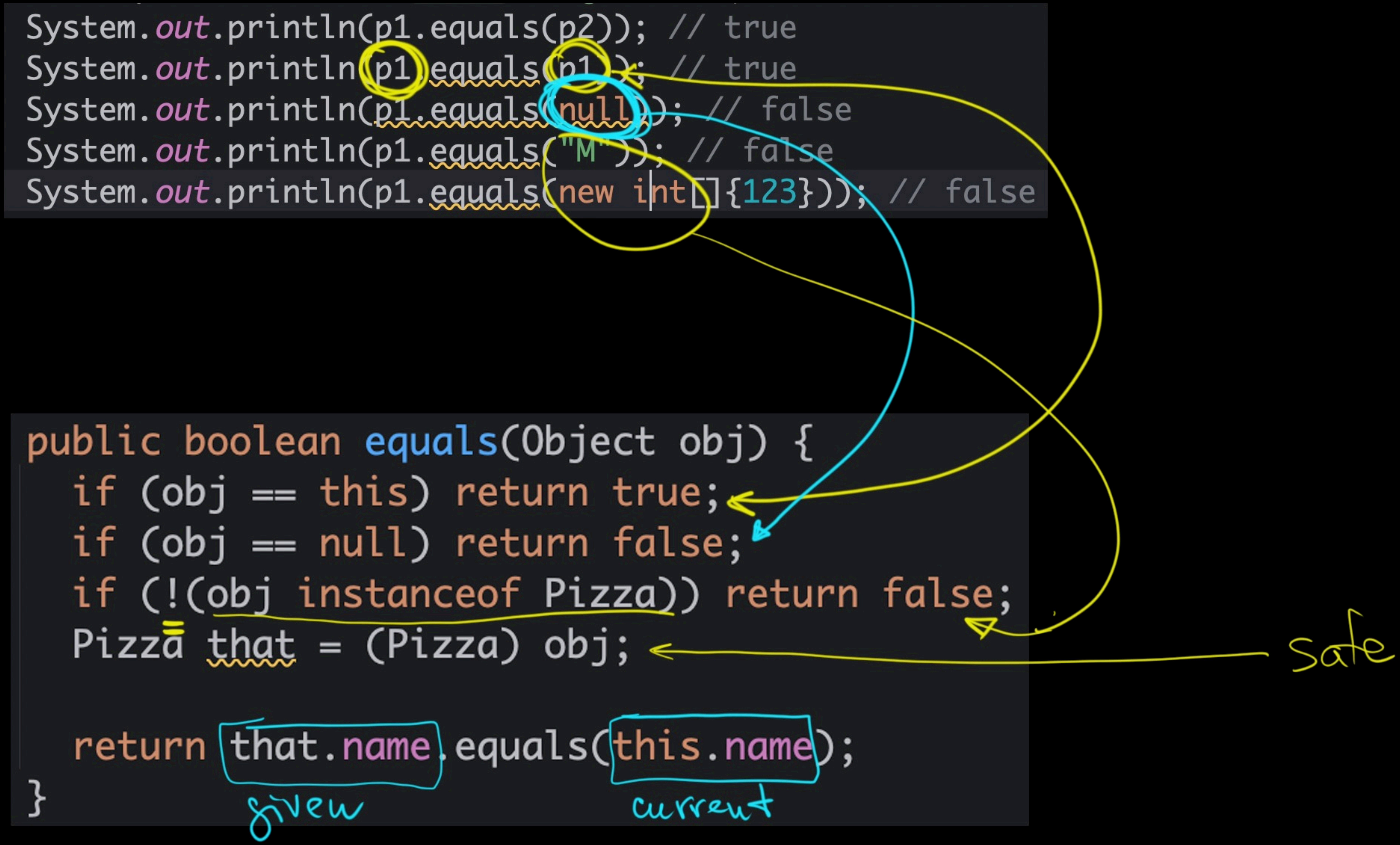
== → primitive, data
→ reference, pointer

1 == ← prim

compare addr in memory

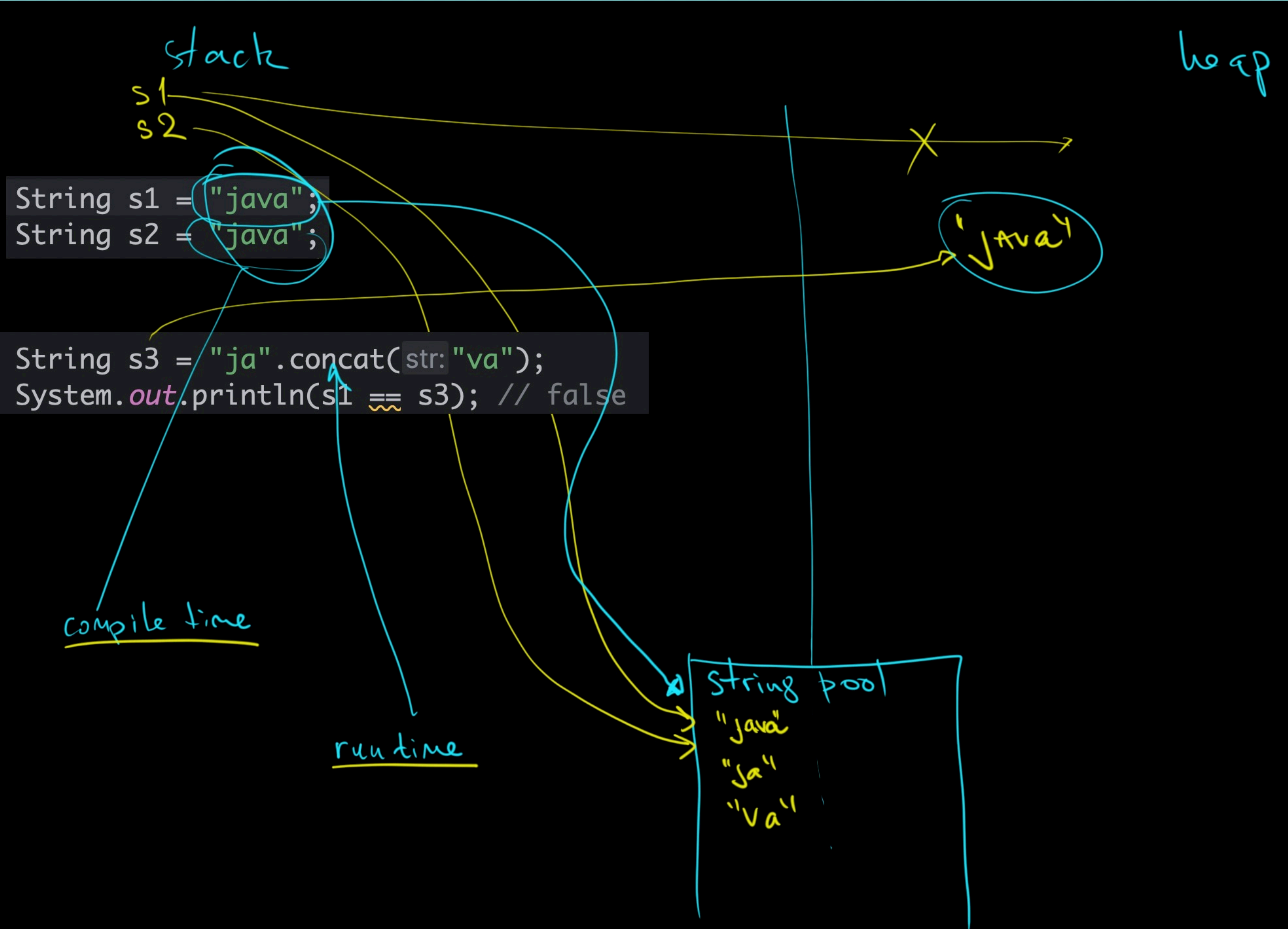
2 ref → .equals( )

user defined



```
System.out.println(p1.equals(p2)); // true
System.out.println(p1.equals(p1)); // true
System.out.println(p1.equals(null)); // false
System.out.println(p1.equals("M")); // false
System.out.println(p1.equals(new int[]{123})); // false
```

```
public boolean equals(Object obj) {
  if (obj == this) return true;
  if (obj == null) return false;
  if (!(obj instanceof Pizza)) return false;
  Pizza that = (Pizza) obj;

  return that.name.equals(this.name);
}
```



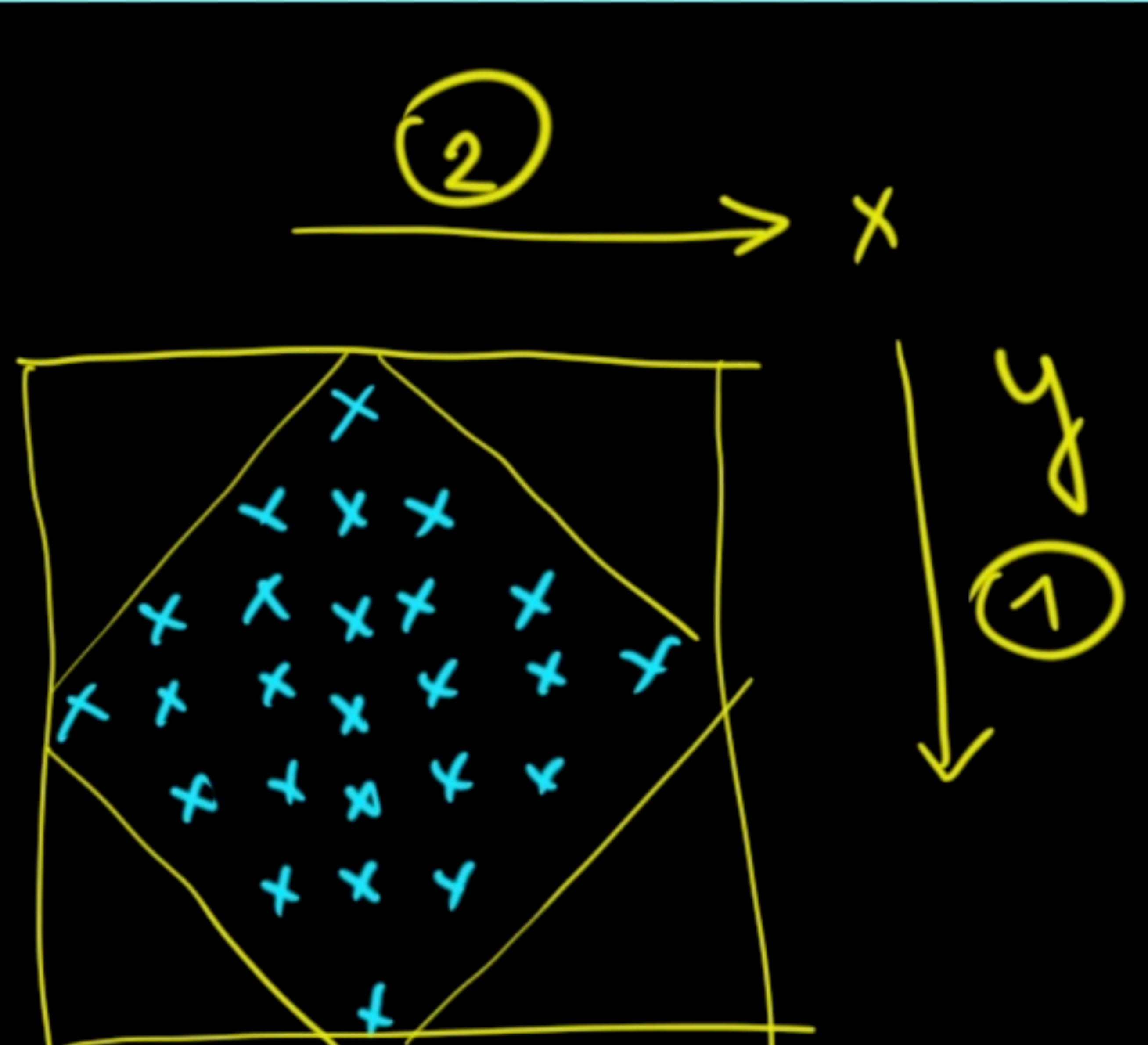
```
String s1 = "java";
String s2 = "java";
```

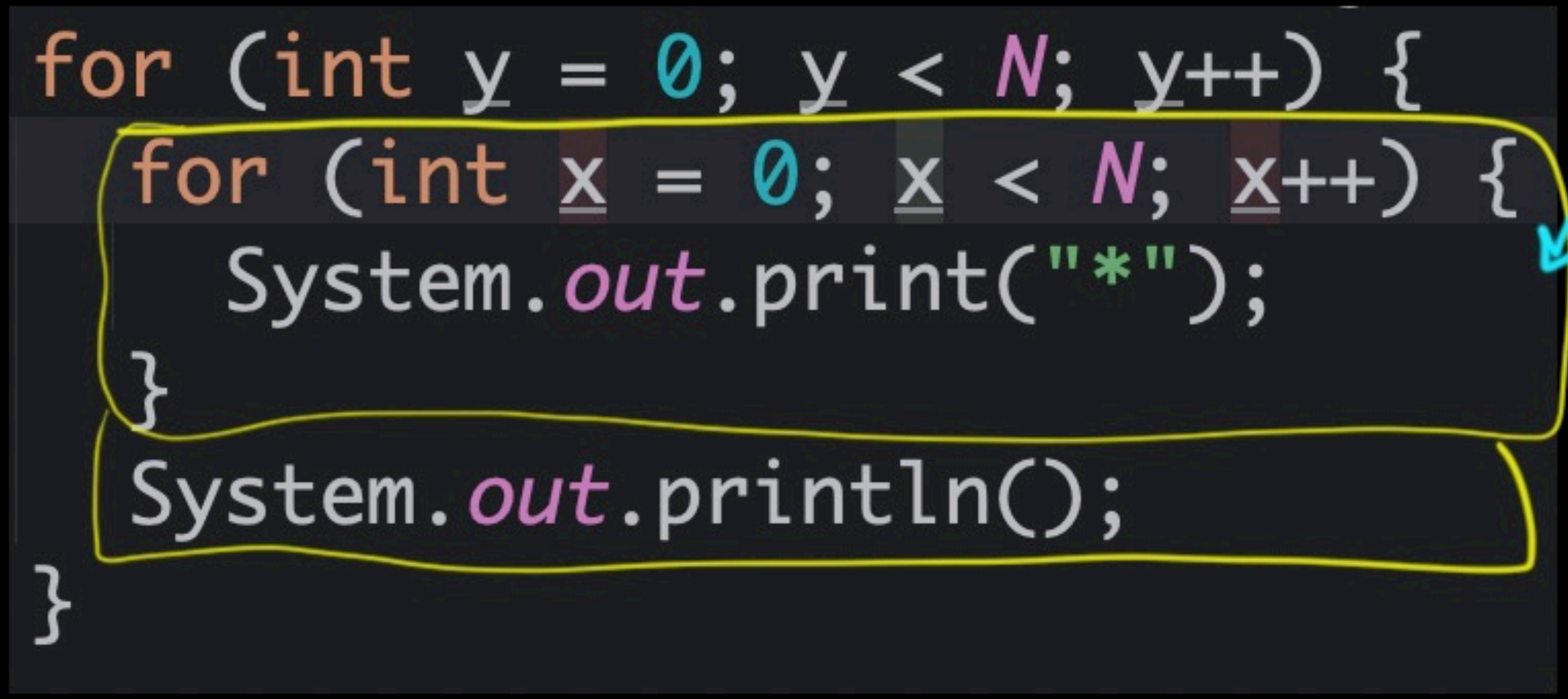
```
String s3 = "ja".concat(str: "va");
System.out.println(s1 == s3); // false
```

```
String s1 = "java";
String s2 = "java";
System.out.println(s1 == s2); // true
```

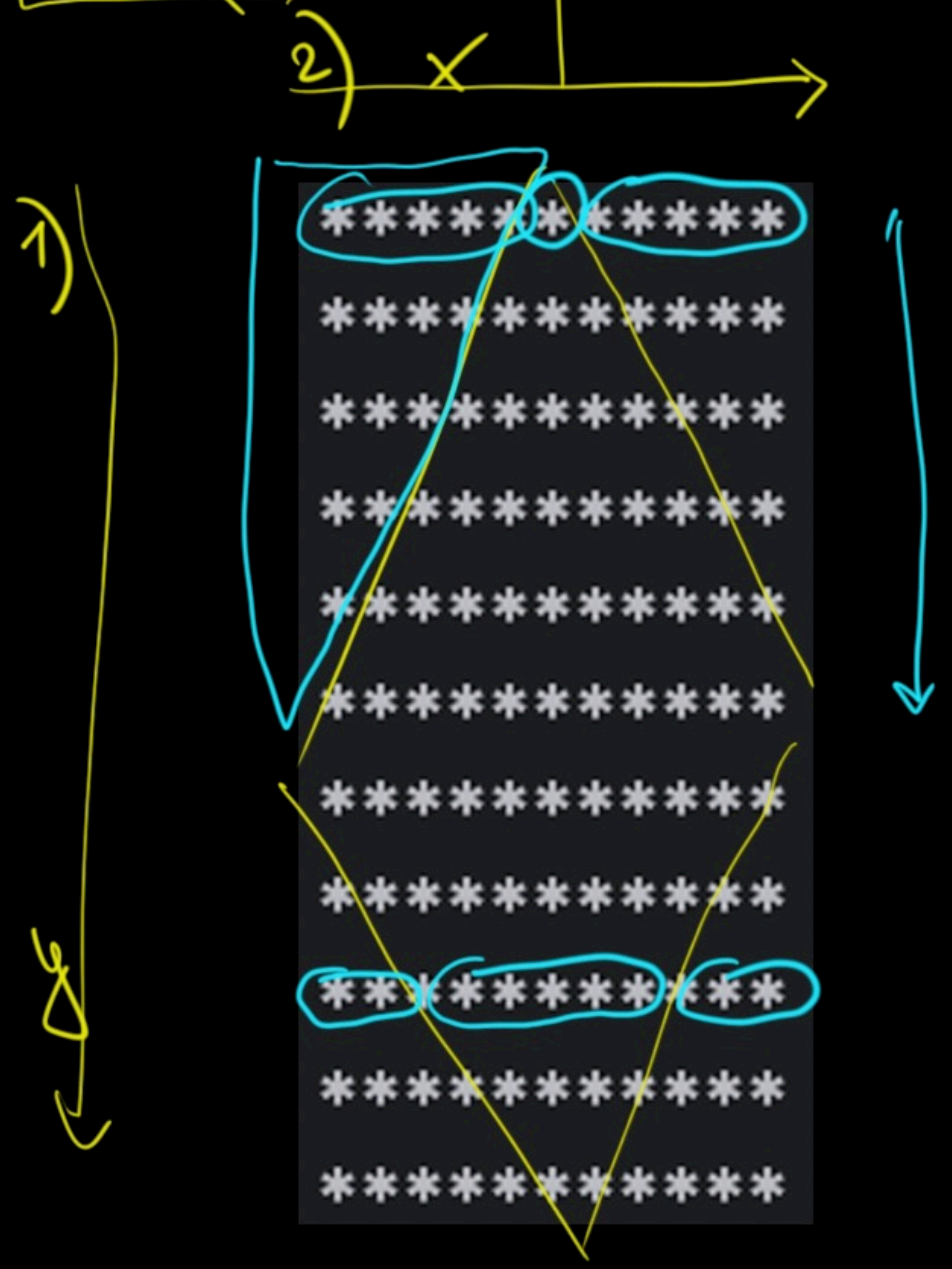
```
String s4 = new String( original: "java");
String s5 = new String( original: "java");
System.out.println(s4 == s5); // false
```

literal, compile time -> string pool
heap
"java"
object creation
runtime -> heap
"java"
String pool
"java"

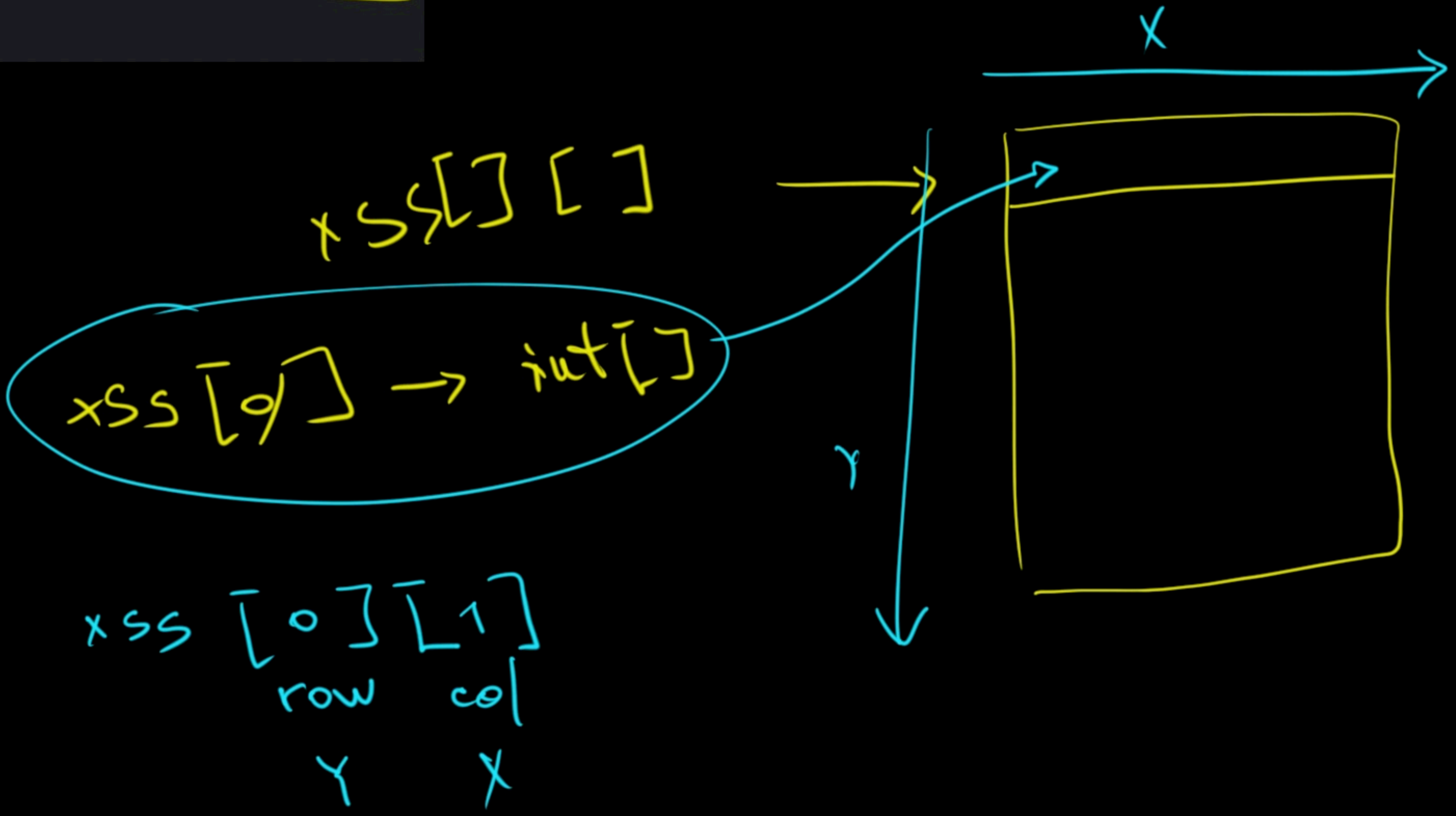


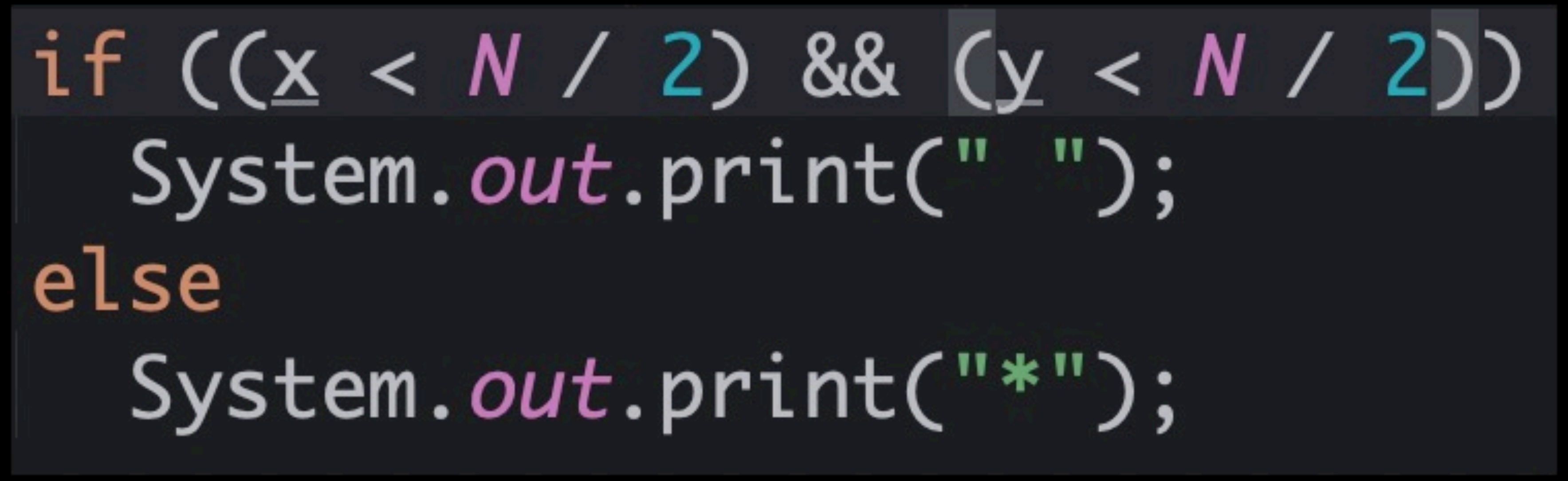


```
for (int y = 0; y < N; y++) {
  for (int x = 0; x < N; x++) {
    System.out.print("*");
  }
  System.out.println();
}
```

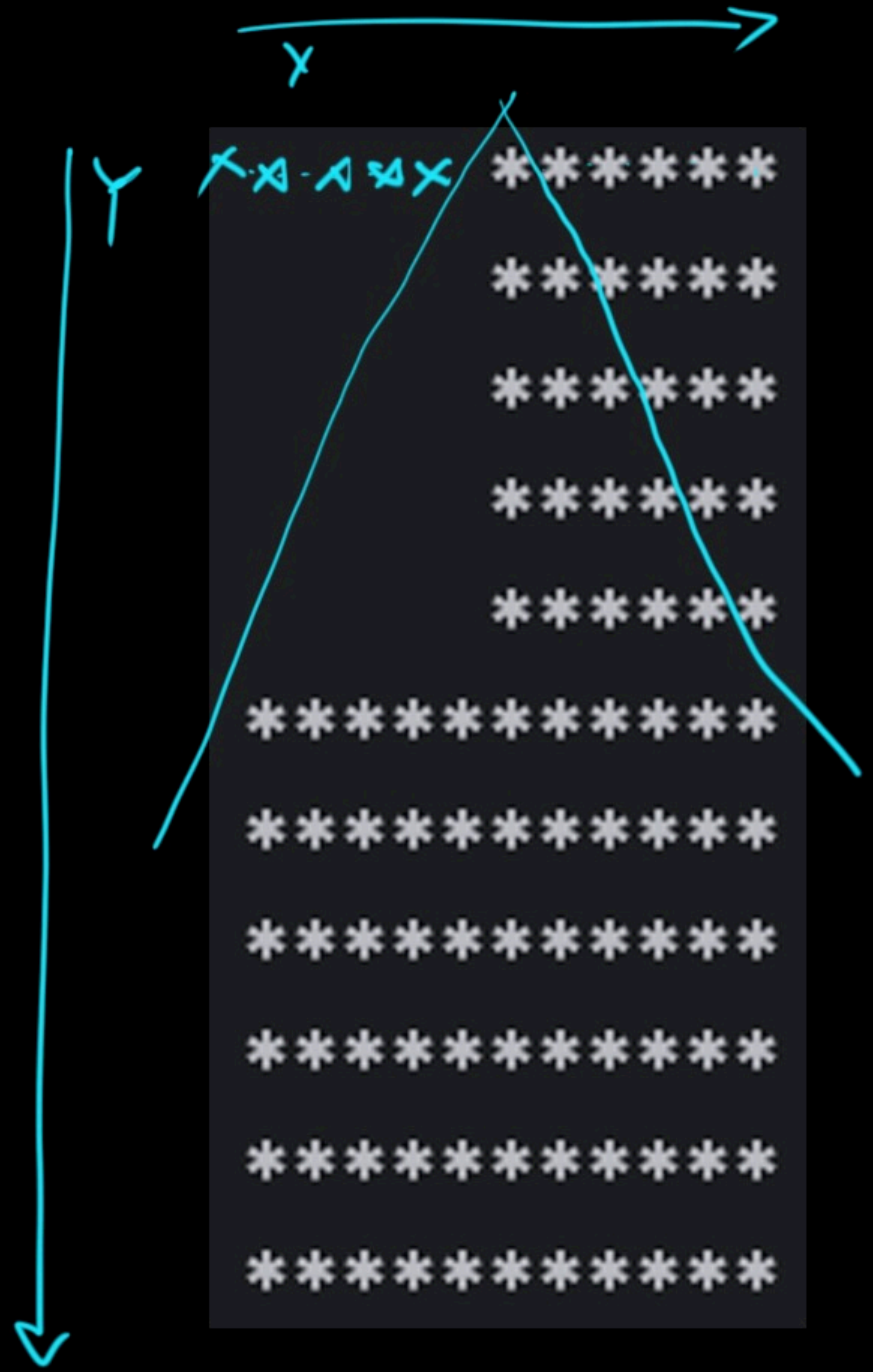


```
***********
***********
***********
***********
***********
***********
***********
***********
***********
***********
***********
```





```
if ((x < N / 2) && (y < N / 2))
  System.out.print(" ");
else
  System.out.print("*");
```

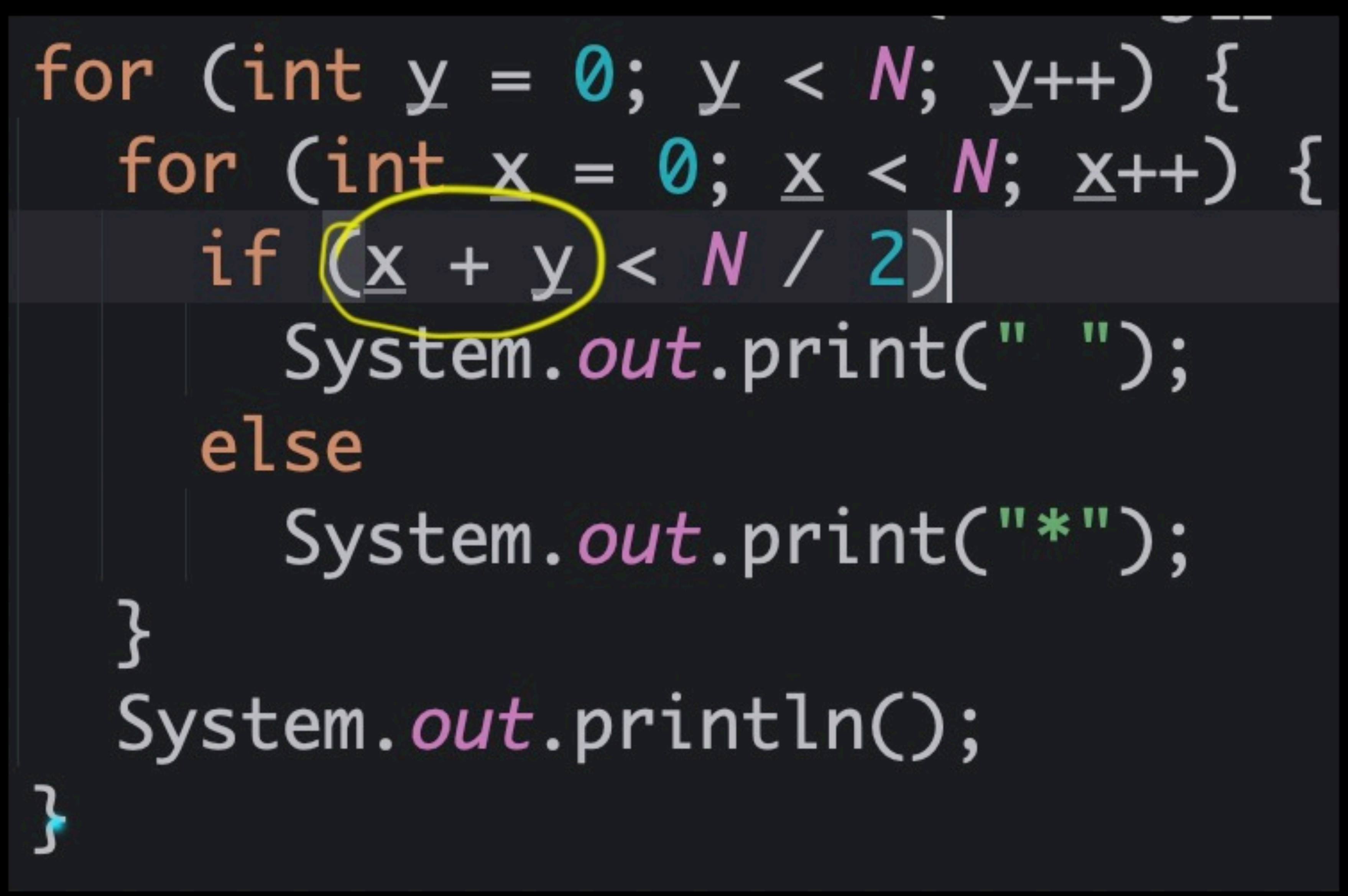


```
      ******
      ******
      ******
      ******
      ******
************
************
************
************
************
************
```

X	Y
0	0
1	0
2	0
3	0
4	0
0	1
1	1
2	1
3	1
0	2
1	2
2	2

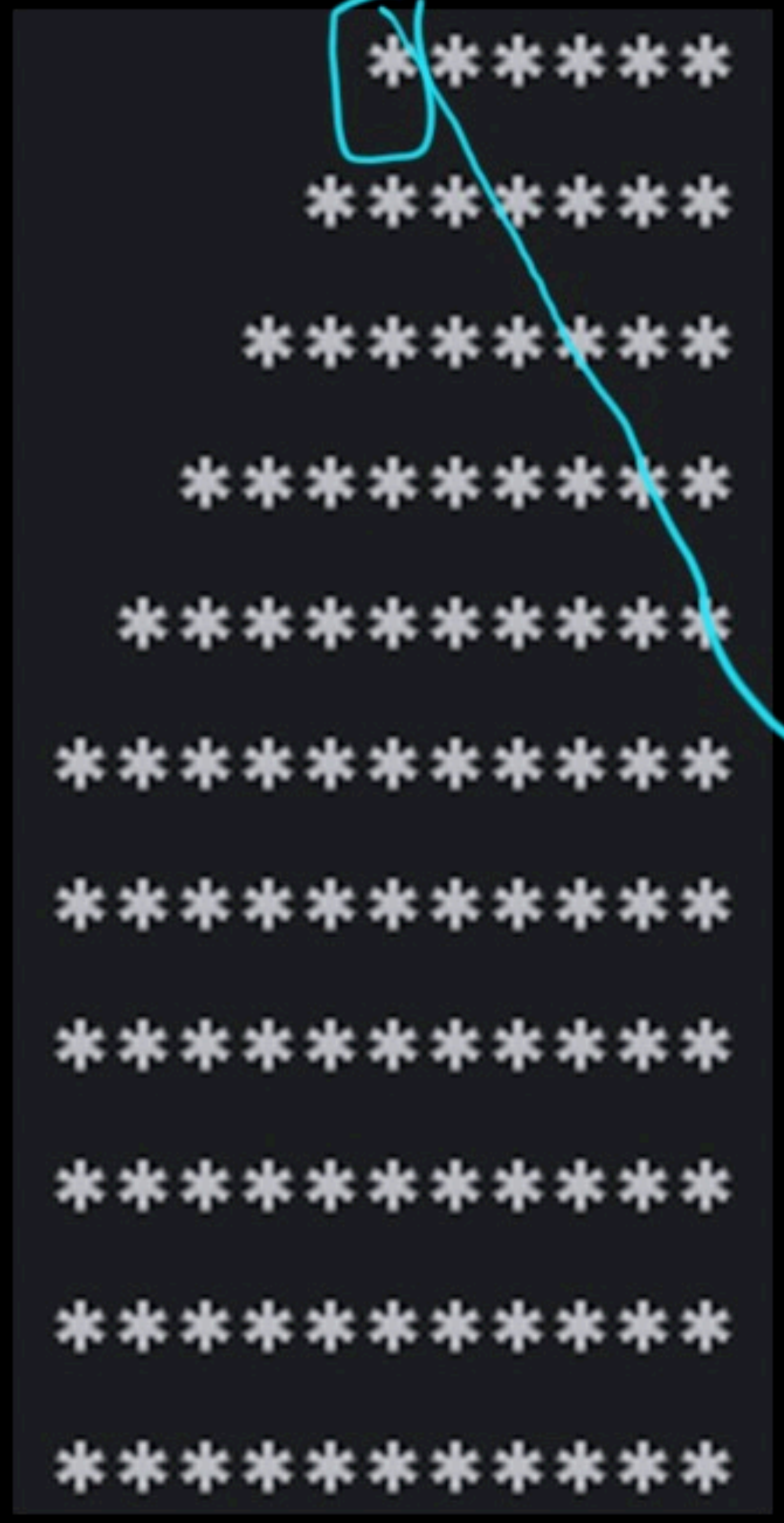
$X+Y < N/2$

$N=11$



```
for (int y = 0; y < N; y++) {
  for (int x = 0; x < N; x++) {
    if (x + y < N / 2)
      System.out.print(" ");
    else
      System.out.print("*");
  }
  System.out.println();
}
```

0 . . . . 4 5 6   10



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

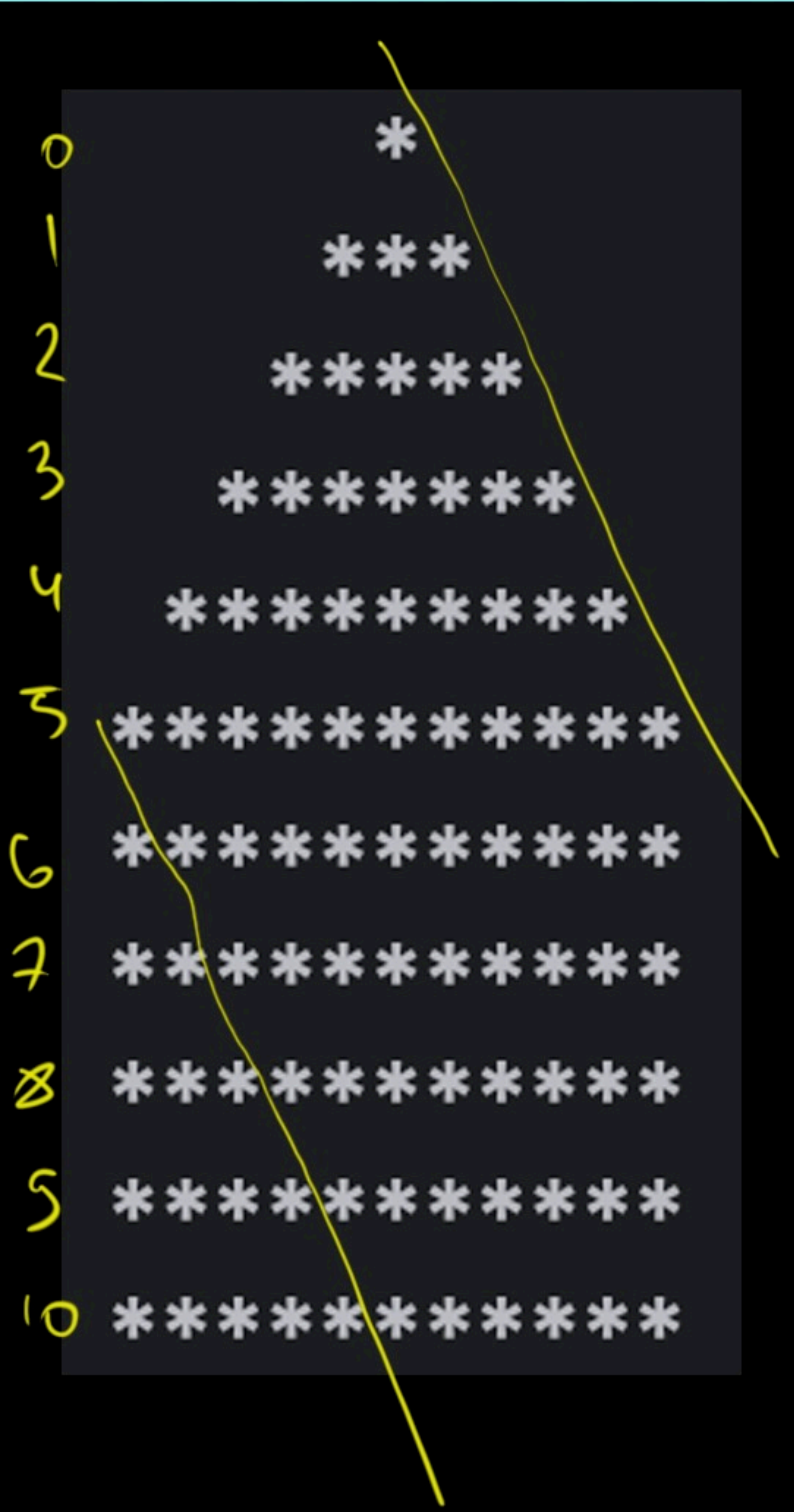
	x_2	X	Y
10	4	6	0
10	3	7	0
10	2	8	0
10	1	9	0
	0	10	0
	3	7	1
	2	8	1
	1	9	1
	0	10	1
	2	8	2
	1	9	2
	0	10	2
	1	9	3
	0	10	3
10	0	10	4

$N-1-x+y < N/2 \quad N=11$

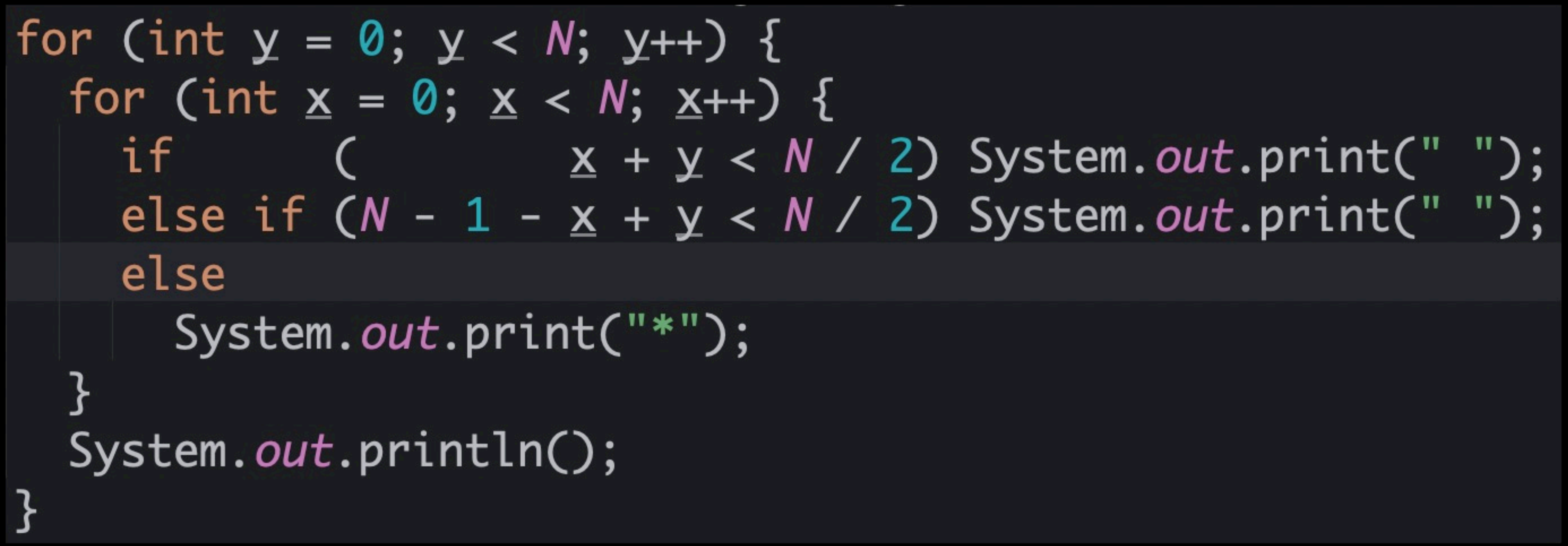
$x_2 = N-1-y$

$4 = 11-1-6$

$3 = 11-1-7$



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



```
for (int y = 0; y < N; y++) {
  for (int x = 0; x < N; x++) {
    if       (         x + y < N / 2) System.out.print(" ");
    else if (N - 1 - x + y < N / 2) System.out.print(" ");
    else
      System.out.print("*");
  }
  System.out.println();
}
```

y=6 x=0

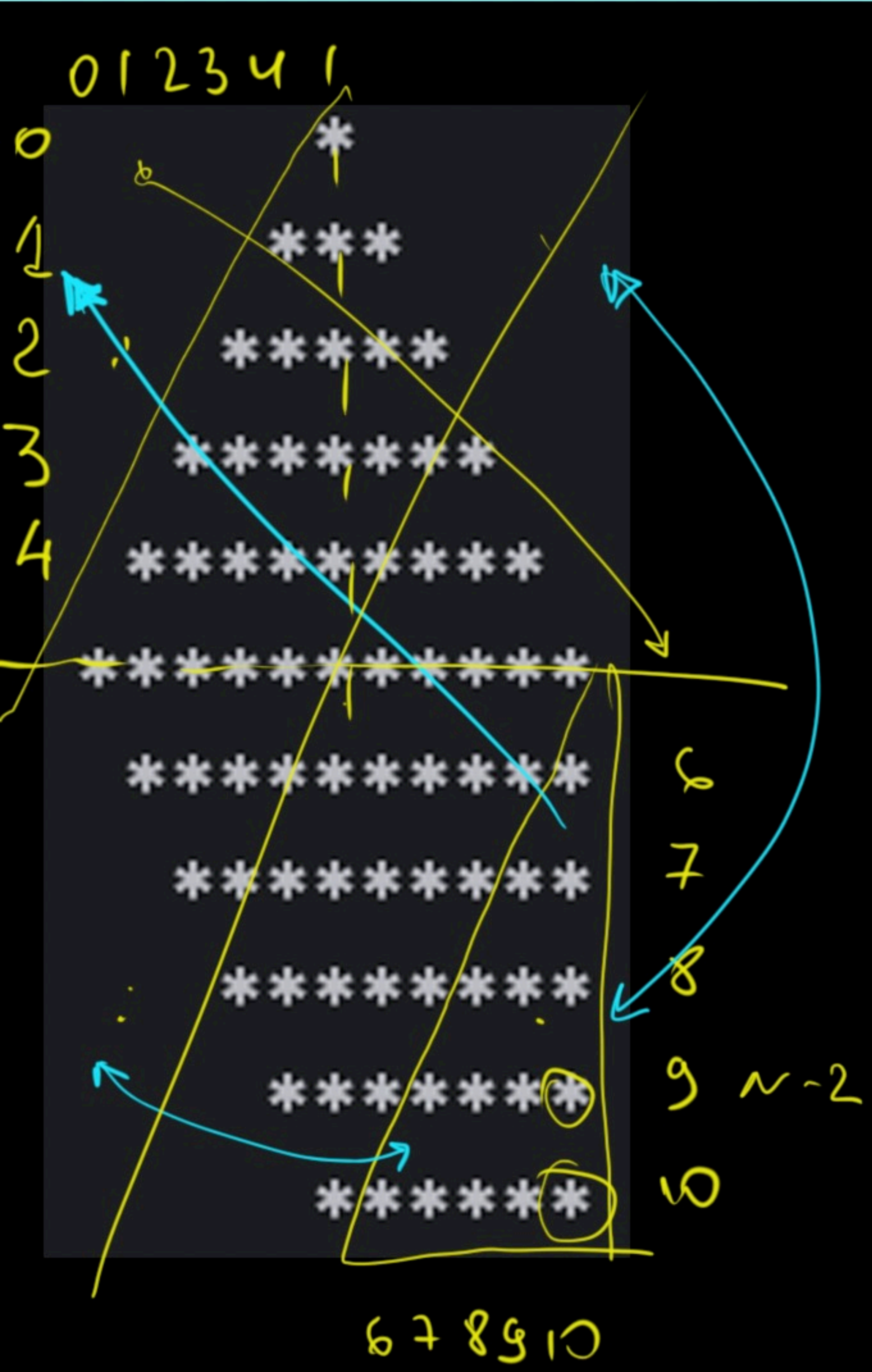
y=7 x=0,1

y=8 x=0,1,2

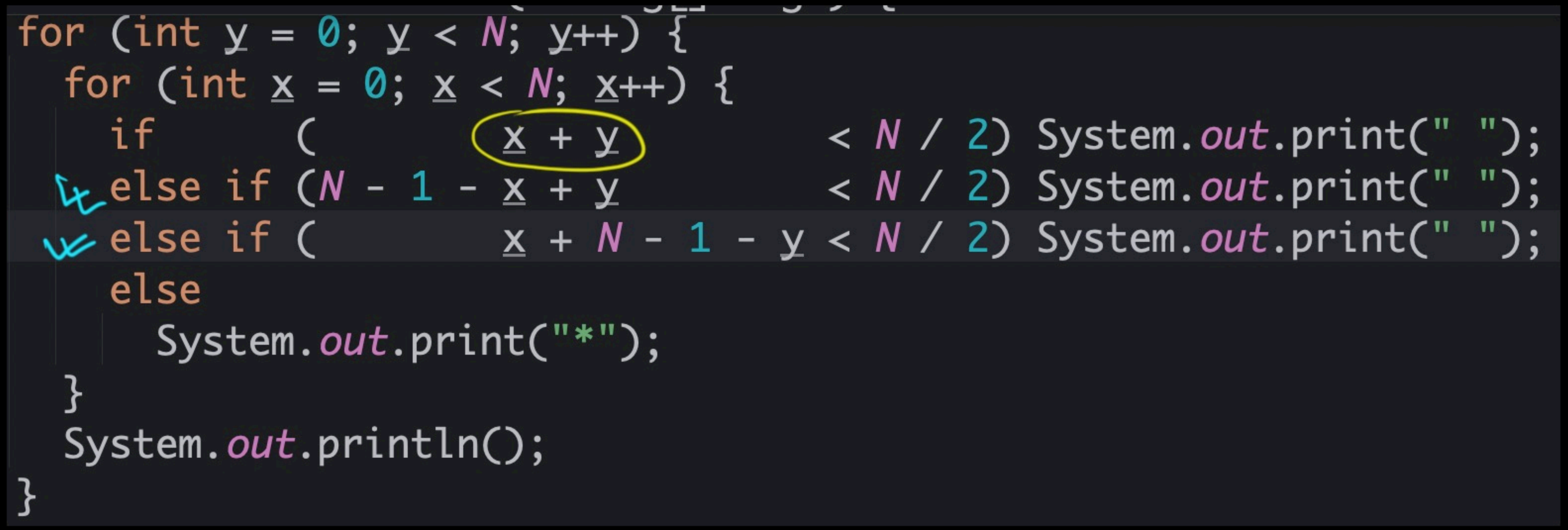
y=9 x=0,1,2,3

y=10 x=0,1,2,3,4

#10

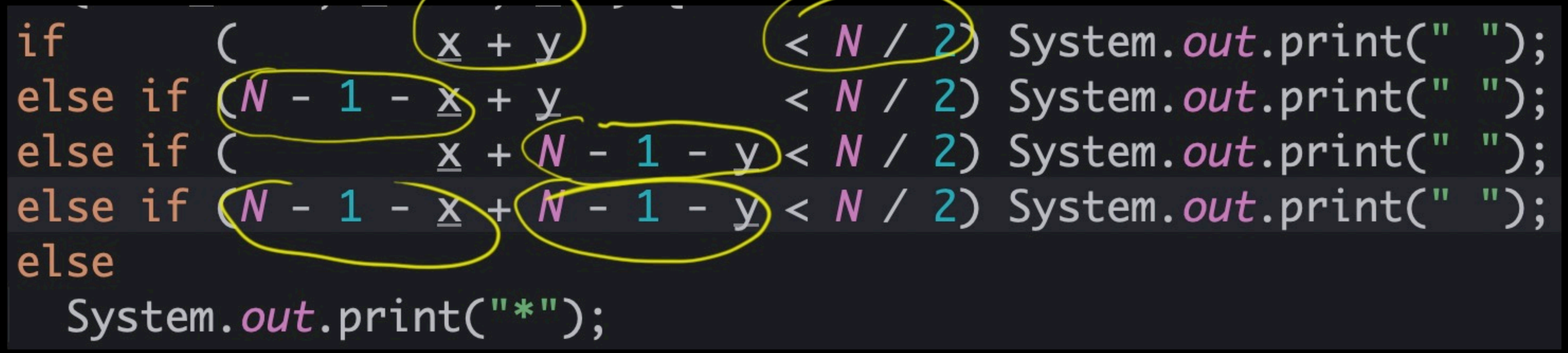


```
     *
    ***
   *****
  *******
 *********
***********
 **********
  *********
   ********
    *******
     ******
```

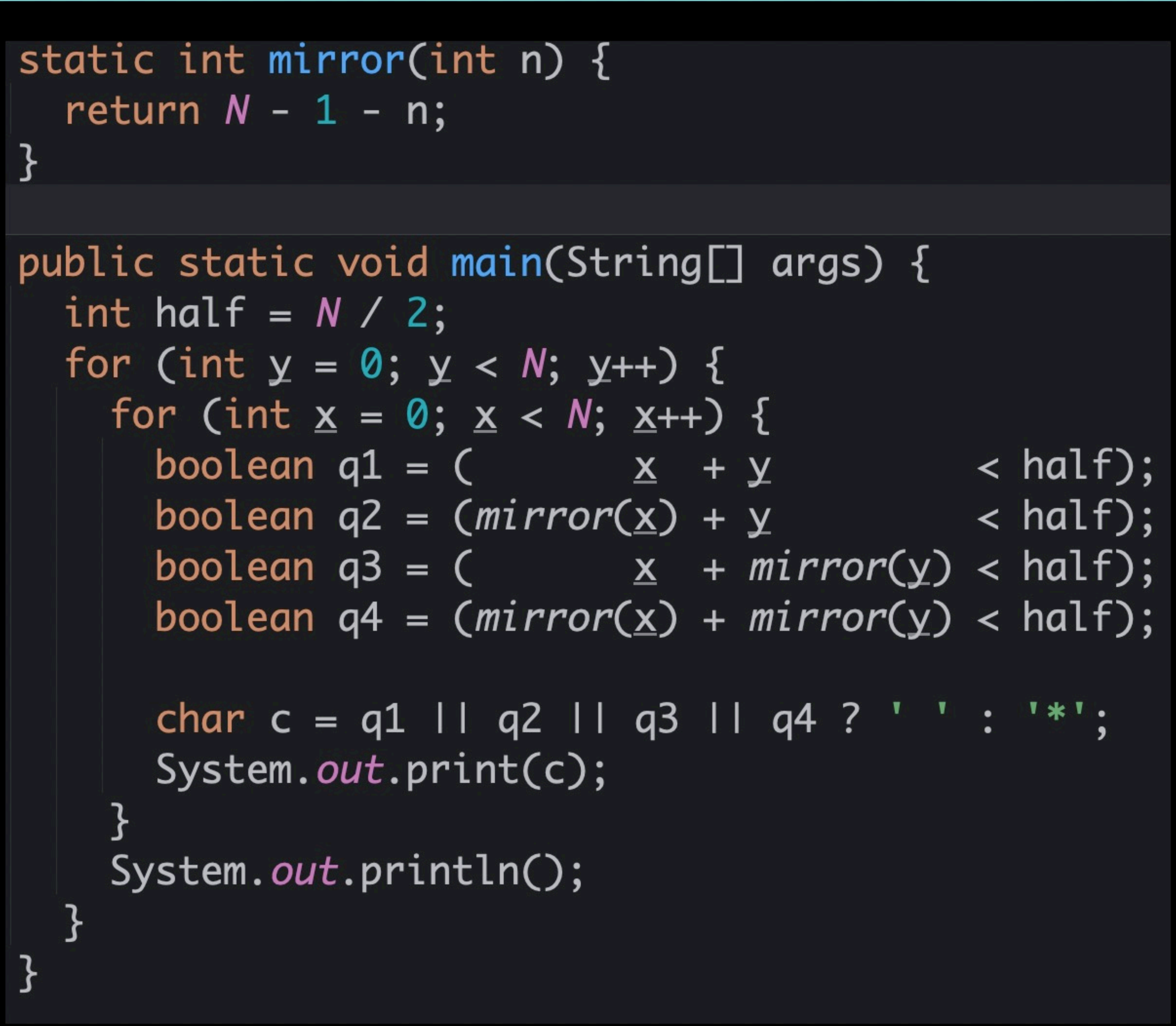


```
for (int y = 0; y < N; y++) {
  for (int x = 0; x < N; x++) {
    if       (          x + y           < N / 2) System.out.print(" ");
    else if (N - 1 - x + y              < N / 2) System.out.print(" ");
    else if (           x + N - 1 - y   < N / 2) System.out.print(" ");
    else
      System.out.print("*");
  }
  System.out.println();
}
```

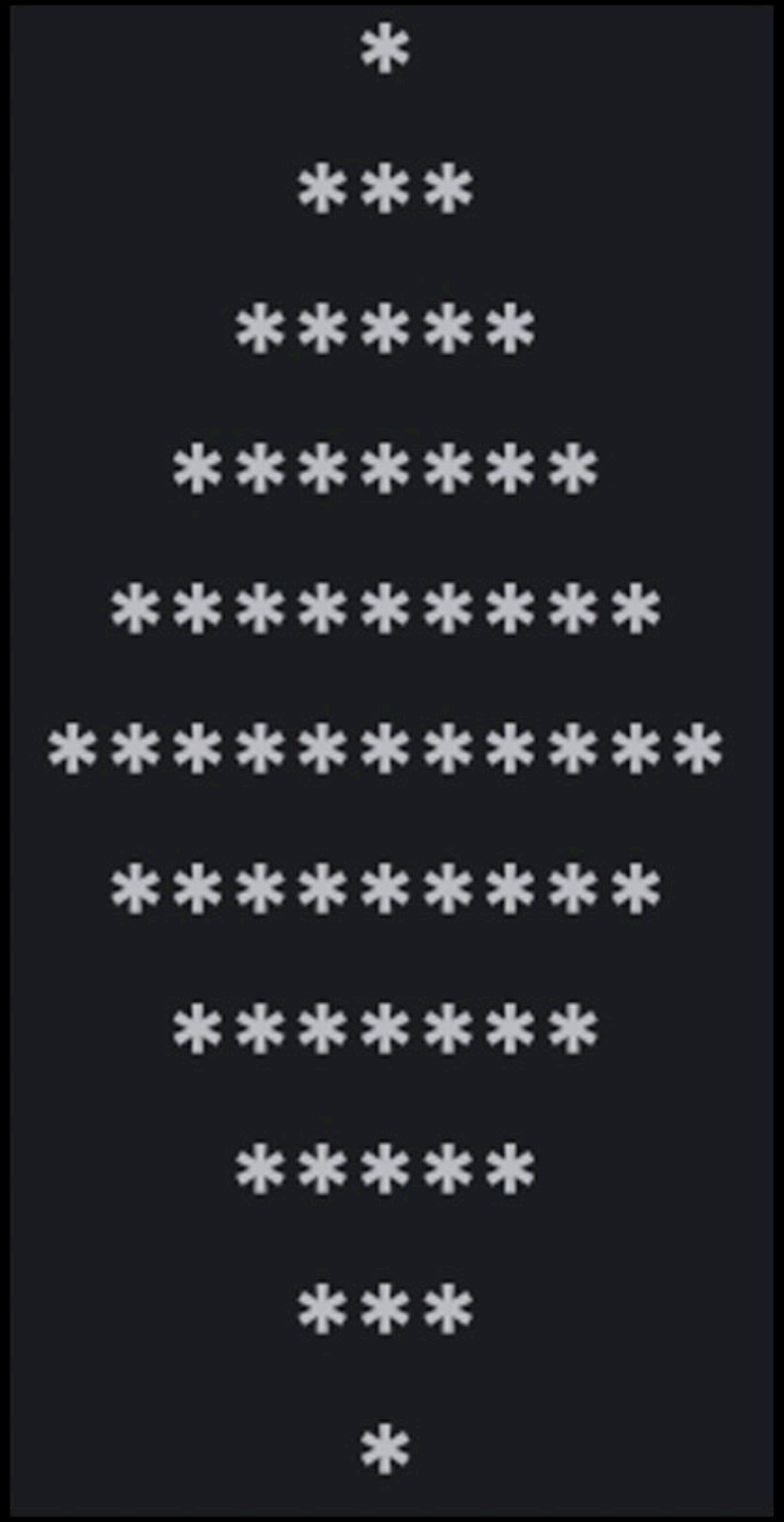
$N-1-x + N-1-y$



```
if       (          x + y                   < N / 2) System.out.print(" ");
else if (N - 1 - x + y                      < N / 2) System.out.print(" ");
else if (           x + N - 1 - y           < N / 2) System.out.print(" ");
else if (N - 1 - x + N - 1 - y              < N / 2) System.out.print(" ");
else
  System.out.print("*");
```



```java
static int mirror(int n) {
  return N - 1 - n;
}

public static void main(String[] args) {
  int half = N / 2;
  for (int y = 0; y < N; y++) {
    for (int x = 0; x < N; x++) {
      boolean q1 = (       x  + y         < half);
      boolean q2 = (mirror(x) + y         < half);
      boolean q3 = (       x  + mirror(y) < half);
      boolean q4 = (mirror(x) + mirror(y) < half);

      char c = q1 || q2 || q3 || q4 ? ' ' : '*';
      System.out.print(c);
    }
    System.out.println();
  }
}
```



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