Example 1. $n = 100$

```
1  int count = 0;
2
3  void example1(int n) {
4      for (int i = 0; i < n; i++)
5          count++;
6  }
```

$O(n)$

count = 100

Example 2.

```
1  void example2(int n) {
2      for (int i = 0; i < n; i++) {
3          for (int j = 0; j < n; j++)
4              count++;
5      }
6  }
```

$O(n)$ $O(n)$ $O(n \cdot n) = O(n^2)$

count = $100^2 = 10\,000$

Example 3.

```
1  void example3(int n) {
2      for (int i = 0; i < n; i++) {
3          for (int j = i; j < n; j++)
4              count++;
5      }
6  }
```

$O(n^2)$

count = $\frac{100(100+1)}{2} = 5050$

i	# times inner loop runs
0	n
1	$n-1$
2	$n-2$
$\vdots$	$\vdots$
$n-1$	1

$$\sum_{i=1}^{n} i = \frac{n(n+1)}{2} = \frac{n^2}{2} + \frac{n}{2} = O(n^2)$$

Example 4.

```
1 void example4(int n) {
2     for (int i = 0; i * i < n; i++)
3         count++;
4 }
```

$i^2 < n$, $i < \sqrt{n}$ } $\sigma(\sqrt{n})$

count = $\sqrt{100} = 10$

Example 5.

```
1 void example5(int n) {
2     for (int i = 1; i < n; i += i)
3         count++;
4 }
```

$2^i < n$, $i < \log_2 n$ } $\sigma(\log n)$

$i = 1, 2, 4, 8, \ldots$ (doubling)

count = $\lceil \log_2 100 \rceil = \lceil 6.\ldots \rceil = 7$

$2^6 = 64$

$2^7 = 128$

2^0 2^1 2^2 2^3 2^4 2^5 2^6

1 2 4 8 16 32 64

Example 6.

```
1 void example6(int n) {
2     for (int i = n * n * n; i > 0; i /= 3)
3         count++;
4 }
```

$\sigma(\log_3(n^3))$

$= \sigma(3 \log_3 n)$

$= \sigma(\log n)$

count = $\lceil \log_3 100^3 \rceil = 13$

Example 7.

```
1 void example7(int n) {
2     for (int i = 0; i < n; i += 5)
3         count++;
4 }
```

$\sigma(n/5) = \sigma(n)$

count = $\frac{100}{5} = 20$

$n = 20$ 0 5 10 15

$20/5 = 4$

Example 8.

```
1  void example8(int n) {
2      int m = 0;
3      int i = n;
4      while (i > 0) {
5          i /= 4;
6          m++;
7      }
8
9      for (i = 0; i < n * m; i++)
10         count++;
11 }
```

① $O(\log_4 n) = O(\log n)$

② $O(nm)$

Take slowest $O(n \log n)$

count = nm

$= n \log_4 n$

$= O(n \log n)$

$= 100 \lceil \log_4 100 \rceil = 100 \cdot \lceil 3.. \rceil = 100 \cdot 4 = 400$

$4^3 = 64$

Example 9.

```
1  void example9(int n) {                          ~ O(n^2)
2      for (int i = 0; i < n * n; i++) {
3          for (int j = 0; j * j * j < n; j++)
4              count1++;
5          for (int j = 1; j < n; j *= 2)
6              count2++;
7      }
8  }
```

Take slowest $O(\sqrt[3]{n})$

① $j^3 < n$, $j < \sqrt[3]{n}$ → $O(\sqrt[3]{n})$

② $O(\log_2 n) = O(\log n)$

Total = $O(\sqrt[3]{n} \cdot n^2) = O(n^{2 1/3})$

count 1 = $\lceil \sqrt[3]{100} \rceil \, 100^2 = 50000$

count 2 = $\lceil \log_2 100 \rceil \, 100^2 = 70000$