

Snippet 1

1. Example [3,6,2,1,4]:
 1. $\langle i=0 \rangle$: $j=1$, $a[i]=3 < 6=a[j]$;
 $j=2$, $a[i]=3 > 2=a[j]$, the array is updated to [2,6,3,1,4];
 $j=3$, $a[i]=2 > 1=a[j]$, the array is updated to [1,6,3,2,4];
 $j=4$, $a[i]=1 < 4=a[j]$.
 2. $\langle i=1 \rangle$: $j=2$, $a[i]=6 > 3=a[j]$, the array is updated to [1,3,6,2,4];
 $j=3$, $a[i]=3 > 2=a[j]$, the array is updated to [1,2,6,3,4];
 $j=4$, $a[i]=2 < 4=a[j]$.
 3. $\langle i=2 \rangle$: $j=3$, $a[i]=6 > 3=a[j]$, the array is updated to [1,2,3,6,4];
 $j=4$, $a[i]=3 < 4=a[j]$.
 4. $\langle i=3 \rangle$: $j=4$, $a[i]=6 > 4=a[j]$, the array is updated to [1,2,3,4,6].
 5. The nested for-loop stops executing.
The array is updated to [1,2,3,4,6] by this snippet, it resorts the array in increasing order.
2. The basic operation is the comparison inside the nested for-loop. The worst-case inputs for this snippet are arrays in strictly decreasing order. In this case, each time the code checks the if-statement it has to update the array.
3. Let n denote length of array, the nested for-loop executes $n*(n-1)/2$ times. In the worst-case, the comparison causes the array updates $n*(n-1)/2$ times (updated everytime as for-loop runs). Therefore, the run-time of this snippet is $n*(n-1)/2 + n*(n-1)/2 = n*(n-1)$ times, which is $O(n^2)$.

Snippet 2

1. Example $x=5$:
 1. $s = ""$.
 2. $y = 5 \% 2 = 1$, s is updated to "1", x is updated to 2 by integer division.
 3. $y = 2 \% 2 = 0$, s is updated to "01", x is updated to 1.
 4. $y = 1 \% 2 = 1$, s is updated to "101", x is updated to 0; the while-loop stops execution.
 4. $s = "101"$ is returned.The snippet converts the input to its binary form as a string.
2. The basic operation is the condition of the while-loop. There is no worst-case input.
3. Let n be the integer input, the while-loop executes $[(\text{int}) \log_2 n + 1]$ times. Therefore, the run-time of this snippet is $O(\log_2 n)$.

Snippet 3

1. Example $x=3$:
 1. $3 * 3 * 3 = 27$ is returned.The snippet calculates the input's cube.
2. The basic operation is the multiplication. There is no worst-case input.

3. Let n be the integer input, the calculation happens once. Therefore, the run-time of this snippet is $O(1)$.

Snippet 4

1. Example $a = [2, 2, 2, 2, 2, 2]$:
 1. $a[0]$ is updated to $2^0 = 1$.
 2. $a[1]$ is updated to $2^1 = 2$.
 3. $a[2]$ is updated to $2^2 = 4$.
 4. $a[3]$ is updated to $2^3 = 8$.
 5. $a[4]$ is updated to $2^4 = 16$.
 6. $a[5]$ is updated to $2^5 = 32$; for-loop stops.The snippet updates array's each item to $a[i]^i$ where i is the index of the item.
2. The basic operation is the exponentiation inside the for-loop. There is no worst-case input.
3. Let n be the length of the array, then the for-loop executes n times (i starting from 0 to $n-1$). Therefore, the run-time of this snippet is $O(n)$.