Question 1.

(1) [0, 1, 2, 4, 5, 7, 8]

(2) $\theta(n+k)$ or $3n+2k$

(3) $\theta(n+k)$ or $2n+k$

(4) yes (stable)

Question 2.

$$k+1$$

Question 3.

Two answers are possible.

(Most Significant Bit → Least Significant Bit)

12256, 13215, 14562, 15873, 22147, 33579, 57616, 79514

(Least Significant Bit → Most Significant Bit)

22147, 13215, 12256, 79514, 14562, 33579, 57616, 15873

Question 4.

The height of the recursion tree of this algorithm is minimum $\log_{100} n$ and maximum $\log_{\frac{100}{99}} n$.

And the time complexity of the partition at all heights is $\theta(n)$.

Therefore the time complexity is between $n \log_{100} n$ and $n \log_{\frac{100}{99}} n$.

Consequently, the time complexity is $\theta(n \log n)$.

Question 5.

(1) The recursion stops when $\frac{n}{2^i} = k$, that is $i = \log_2 \frac{n}{k}$.

Therefore, the quicksort takes $O(c_1 n \log \frac{n}{k})$.

There are $\frac{n}{k}$ subarrays with size k.

Therefore, the insertionsort takes $\frac{n}{k} O(c_2 k^2) = O(c_2 nk)$.

Consequently, the total time complexity of optimized-quicksort is $O(c_1 n \log \frac{n}{k} + c_2 nk)$.

(2) Condition: $c_1 n \log \frac{n}{k} + c_2 nk < c_1 n \log n$

In short: $c_2 k < c_1 \log k$