Task

For this lab you will determine the Big-O of several methods. For each one, provide the Big-O complexity, along with a short justification. Also, be sure to indicate what ?n? refers to.
You can determine the complexities by counting steps, using reasoning skills, or by running the code in order to understand how the methods are working.

1. O(n) time complexity worst case, with O(1) as best case. It look's through each entry of the array until it finds a the matching number.

2. O(1) time complexity. This is a constant time operation, which just checks and compares.

3. O(n^2) time complexity. I see a single nested for-loop, which means there is a for loop inside another one.

4. O(n^1/2) time complexity. A finding prime function iterates only up to the function's square root.

5. O(10^n) time complexity. Function generates 10 different codes for n amount of slots. n = 1 is 0..9, n = 2 is 0..99 and it follows with every other integer n.