**1.Easy\03.Largest_odd_number_in_string.cpp**

```
/*
Question:
You are given a string num, representing a large integer. Return the largest-valued odd integer (as a string) that is a non-empty substring of num, or an empty string "" if no odd integer exists.
A substring is a contiguous sequence of characters within a string.

Approach:
1. Iterate through the string from the last character.
2. Check if the current character is odd.
3. If it is odd, return the substring from the beginning of the string to the current character index.
4. If no odd number is found, return an empty string.

Time Complexity: O(N), where N is the length of the input string num.
- We iterate through the string once to find the largest odd number.

Space Complexity: O(1)
- We use constant space to store the result and iterate through the string.

Code:
*/

string largestOddNumber(string num) {
    // Check the first odd number from last
    for (int i = num.size() - 1; i >= 0; i--) {
        if ((num[i] - '0') % 2 != 0)
            return num.substr(0, i + 1);
    }
    return "";
}

```