```cpp
#include <iostream>
#include <string>
using namespace std;

// Function to add 1 to a binary number represented as a string
string addOneToBinary(string& binaryNumber) {
    string result = binaryNumber; // Creates a copy of the input binary number
    int carry = 1;  // Initializes the carry to 1 to add 1

    // Traverses the binary number from right to left
    for (int i = binaryNumber.length() - 1; i >= 0; i--) {
        if (binaryNumber[i] == '0') {
            // If the current digit is '0', adds the carry
            result[i] = '1';
            carry = 0;  // Resets the carry
            break;
        } else {
            // If the current digit is '1', sets it to '0' and continues
            result[i] = '0';
        }
    }

    // If there's still a carry, adds '1' at the beginning of the result
    if (carry == 1) {
        result = '1' + result;
    }

    return result; // Returns the result
}

// Function to generate all possible combinations of characters in a string
void generateCombinations(string& str) {
    int n = str.length(); // Gets the length of the input string

    // The total number of combinations is 2^n
    int totalCombinations = 1 << n;

    for (int i = 1; i < totalCombinations; i++) {
        string combination; // Creates an empty string to store the current combinati

        for (int j = 0; j < n; j++) {
            // Checks if the j-th bit of i is set
            if (i & (1 << j)) {
                combination += str[j];
                if (j < n - 1) {
                    combination += ' '; // Add a space between characters
                }
            }
        }

        cout << combination << endl; // Outputs the generated combination
    }
}


int main() {
    string binaryNumber; // Declares a string to store the binary number

    // Part (a): Adding 1 to a binary number
    cout << "Enter a binary number: "; // Prompts the user for input
    cin >> binaryNumber; // Reads the binary number from the user
    string result = addOneToBinary(binaryNumber); // Calls the addOneToBinary functio
    cout << "Result: " << result << endl; // Outputs the result

    // Part (b): Generating all possible combinations
    string inputString; // Declares a string to store the input string
    cout << "Enter a string to generate combinations: "; // Prompts the user for inpu
    cin >> inputString; // Reads the input string from the user
    generateCombinations(inputString); // Calls the generateCombinations function

    return 0; // Returns 0 to indicate successful execution
}

```