

## Day – 57 of the #101 days of the coding challenge-----

### Problem:-

Given two binary strings a and b, return their sum as a binary string.

#### Example 1:

**Input:** a = "11", b = "1"

**Output:** "100"

#### Example 2:

**Input:** a = "1010", b = "1011"

**Output:** "10101"

### Solution:-

```
string addBinary(string a, string b) {  
  
    if(a.length() > b.length())  
    {  
        return addBinary(b, a); // after swapping the string b will be greater.  
    }  
  
    int diff = b.length() - a.length(); // finding the different so that we can  
    add the extra zeros  
  
    string padding;  
    // adding zero's for equal length of the both binaries  
    for(int i = 0; i<diff; i++)  
    {  
        padding.push_back('0');  
    }  
  
    string res; // for storing the result  
    char carry = '0';  
    // we know length of a string is less as per difference then need to add the  
    extra zeros.  
    a = padding + a;
```

```

// adding the binaries number of a and b
for(int i = a.length()-1; i>=0; i--)
{
    if(a[i] == '1' && b[i] == '1')
    {
        if(carry == '1')
            res.push_back('1'), carry = '1';
        else
            res.push_back('0'), carry = '1';
    }

    else if(a[i] == '0' && b[i] == '0')
    {
        if(carry == '1')
            res.push_back('1'), carry = '0';
        else
            res.push_back('0'), carry = '0';
    }

    if(a[i] != b[i])
    {
        if(carry == '1')
            res.push_back('0'), carry = '1';
        else
            res.push_back('1'), carry = '0';
    }

}
// if in the last there is any carry then it will be added on it
if(carry == '1')
    res.push_back('1');

// reversing the bit
reverse(res.begin(), res.end());

return res;
}

```

✓ Accepted

📖 Editorial

+ Solution

🕒 Runtime

Details

**0** ms

**Beats 100.00%** of users with C++

💾 Memory

Details

**7.22** MB

**Beats 34.47%** of users with C++