Given a binary string, count number of substrings that start and end with 1.

Given a binary string, count number of substrings that start and end with 1. For example, if the input string is “00100101”, then there are three substrings “1001”, “100101” and “101”.

Source: Amazon Interview Experience | Set 162

Difficulty Level: Rookie

We strongly recommend to minimize your browser and try this yourself first.

A **Simple Solution** is to run two loops. Outer loops picks every 1 as starting point and inner loop searches for ending 1 and increments count whenever it finds 1.

```
// A simple C++ program to count number of substrings starting and ending
// with 1
#include<iostream>
using namespace std;

int countSubStr(char str[])
{
   int res = 0;  // Initialize result

   // Pick a starting point
   for (int i=0; str[i] !='\0'; i++)
   {
        if (str[i] == '1')
        {
            // Search for all possible ending point
            for (int j=i+1; str[j] !='\0'; j++)
               if (str[j] == '1')
                  res++;
        }
   }
   return res;
}

// Driver program to test above function
int main()
{
  char str[] = "00100101";
  cout << countSubStr(str);
  return 0;
}
```

Run on IDE

Output:

```
3
```

Time Complexity of the above solution is $O(n^2)$. We can find count **in O(n) using a single traversal** of input string. Following are steps.

a) Count the number of 1's. Let the count of 1's be m.

b) Return m(m-1)/2

The idea is to count total number of possible pairs of 1's.

```
// A O(n) C++ program to count number of substrings starting and ending
// with 1
#include<iostream>
using namespace std;

int countSubStr(char str[])
{
   int m = 0; // Count of 1's in input string

   // Travers input string and count of 1's in it
   for (int i=0; str[i] !='\0'; i++)
   {
        if (str[i] == '1')
           m++;
   }

   // Return count of possible pairs among m 1's
   return m*(m-1)/2;
}

// Driver program to test above function
int main()
{
  char str[] = "00100101";
  cout << countSubStr(str);
  return 0;
}
```

Run on IDE

Output:

```
3
```

This article is contributed by **Shivam**. Please write comments if you find anything incorrect, or you want to share more information about the topic discussed above

24 Comments Category: Arrays

Related Posts:

- Longest Span with same Sum in two Binary arrays
- Count Inversions of size three in a give array
- Find the subarray with least average
- Count triplets with sum smaller than a given value
- Find zeroes to be flipped so that number of consecutive 1's is maximized
- Reorder an array according to given indexes
- Find maximum value of Sum(i*arr[i]) with only rotations on given array allowed
- Find maximum average subarray of k length

(Login to Rate and Mark)

2.1 Average Difficulty : **2.1/5.0**
Based on **8** vote(s)

Add to TODO List
Mark as DONE

Writing code in comment? Please use code.geeksforgeeks.org, generate link and share the link here.