```
        ll count = even + odd;
        return count;
    }

    //get longest palindrome subString
    string getLongestPalindrome(){
        int oddIndex = 0 , evenIndex = 0;
        for (int i = 0; i < n; i++){
            if(d1[i] > d1[oddIndex]) oddIndex = i;
            if(d2[i] > d2[evenIndex]) evenIndex = i;
        }
        string even = s.substr(evenIndex-d2[evenIndex] , d2[evenIndex]*2);
        string odd = s.substr(oddIndex-d1[oddIndex]+1 , d1[oddIndex]*2-1);
        return even.size() ⩾ odd.size() ? even : odd;
    }
};
```