Daily Challenge **Happy Coding from necse** **SKILLRACK**

Smallest Substring - All Distinct Letters

A string value S containing only alphabets (both lower and upper case) is passed as input to the program. The program must print the size of the smallest substring which contains all the distinct alphabets in the string S.

Input Format:
First line contains S.

Output Format:
First line contains the size of the smallest substring which contains all the distinct characters in the string S.

Boundary Conditions:
1 <= Length of string S <= 1000

Note:
a and A (lower and upper case letters are considered different). Hence the string aAaaA contains 2 distinct alphabets.

Example Input/Output 1:
Input:
abcdfffccbbfeaad

Output:
8

Explanation:
The smallest sub string that contains all the distinct letters is **cbbfeaad** whose length is 8.

Example Input/Output 2:
Input:
klLDKFFKpPoOGAFSGxxkllhaJSHJAlqadyASFLHJASHFA

Output:
32

Explanation:
The smallest sub string that contains all the distinct letters is **LDKFFKpPoOGAFSGxxkllhaJSHJAlqady** whose length is 32.

Max Execution Time Limit: 5000 millisecs

Ambiance

Java (12.0)

```java
import java.util.*;
public class Hello {

    public static void main(String[] args) {

        Scanner sc = new Scanner(System.in);

        String s = sc.nextLine();

        Map<Character,Integer> ht = new LinkedHashMap<>();

        int max = 0;
        int i=0;
        int len = 0;
        char prev = '\0';

        while(i<s.length()){
          //  System.out.println(ht);
            if(ht.containsKey(s.charAt(i)) && s.charAt(i)!=prev){
                max = Math.max(max,len);
                i = ht.get(s.charAt(i))+1;
                ht.clear();
                prev='\0';
                len=0;
            }else{
                ht.put(s.charAt(i),i);
                prev = s.charAt(i);
                len++;
                i++;
            }

        }
        System.out.println(max+"");


    }
}
```

1912067@nec

Please wait while we run the program ....