3. Longest Substring Without Repeating Characters

Given a string `s`, find the length of the **longest substring** without repeating characters.

Constraints:

- `0 <= s.length <= 5 * 10`^{4}
- `s` consists of English letters, digits, symbols and spaces.

Example 1:

```
Input: s = "abcabcbb"
Output: 3
Explanation: The answer is "abc", with the length of 3.
```

Example 2:

```
Input: s = "bbbbb"
Output: 1
Explanation: The answer is "b", with the length of 1.
```

Example 3:

```
Input: s = "pwwkew"
Output: 3
Explanation: The answer is "wke", with the length of 3.
Notice that the answer must be a substring, "pwke" is a subsequence and not a
substring.
```

```cpp
int lengthOfLongestSubstring(string s) {
    int n = s.length(), result = 0, i = 0, j = 0;
    unordered_set<char> st;
    while(i<n && j<n) {
        if(st.find(s[j]) == st.end()) {
            st.insert(s[j]);
            j++;
            result = max(result, j-i);
        } else {
            st.erase(s[i]);
            i++;
        }
    }
    return result;
}
```

#100daysofDSA

/rvislive

Rakesh Vishwakarma