

### 3. Longest Substring Without Repeating Characters

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

#### Example 1:

Input: "abcabcbb"

Output: 3

Explanation: The answer is "abc", with the length of 3.

#### Example 2:

Input: "bbbbbb"

Output: 1

Explanation: The answer is "b", with the length of 1.

#### Example 3:

Input: "pwwkew"

Output: 3

Explanation: The answer is "wke", with the length of 3.

Note that the answer must be a **substring**, "pwke" is a *subsequence* and not a **substring**.

- 先處理字串長度如果小於2的情況
- 接下來從字串的第2個 index 開始往前比對字串 (s)
  - 如果比到重複的字元 (chk) , 就跳過 , 下次從重複的 index 繼續比對 (head)

```
1  #include <stdio.h>
2  #include <stdlib.h>
3
4  int lengthOfLongestSubstring(char * s)
5  {
6      int len, max = 0;
7      char *head = s, *chk;
8
9      if (*s == 0)
10         return 0;
11     if (*(s + 1) == 0)
12         return 1;
13
14     while (*++s != 0) {
15         chk = head;
16         len = 1;
17
18         while (chk < s) {
19             if (*chk++ != *s) {
```

```
20         len += 1;
21     } else {
22         head = chk;
23         break;
24     }
25 }
26
27 if (len > max)
28     max = len;
29 }
30
31 return max;
32 }
```