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: "bbbbb"
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)`

```c
#include <stdio.h>
#include <stdlib.h>

int lengthOfLongestSubstring(char * s)
{
    int len, max = 0;
    char *head = s, *chk;

    if (*s == 0)
        return 0;
    if (*(s + 1) == 0)
        return 1;

    while (*++s != 0) {
        chk = head;
        len = 1;

        while (chk < s) {
            if (*chk++ != *s) {
                len += 1;
            } else {
                head = chk;
                break;
            }
        }

        if (len > max)
            max = len;
    }

    return max;
}
```