

# 【leetcode】2. Add Two Numbers 3. Longest Substring Without Repeating Characters

作者: [jerwang](#)

原文链接: <https://ld246.com/article/1504587798548>

来源网站: [链滴](#)

许可协议: [署名-相同方式共享 4.0 国际 \(CC BY-SA 4.0\)](#)

2: You are given two **non-empty** linked lists representing two non-negative integers. The digits are stored in reverse order and each of their nodes contain a single digit. Add the two numbers and return it as a linked list.

You may assume the two numbers do not contain any leading zero, except the number 0 itself.

**Input:** (2 -> 4 -> 3) + (5 -> 6 -> 4)

**Output:** 7 -> 0 -> 8

某外国小哥的代码，简洁明了，再优化可读性就差了。

```
ListNode *addTwoNumbers(ListNode *l1, ListNode *l2) {
    ListNode preHead(0), *p = &preHead;
    int extra = 0;
    while (l1 || l2 || extra) {
        int sum = (l1 ? l1->val : 0) + (l2 ? l2->val : 0) + extra;
        extra = sum / 10;
        p->next = new ListNode(sum % 10);
        p = p->next;
        l1 = l1 ? l1->next : l1;
        l2 = l2 ? l2->next : l2;
    }
    return preHead.next;
}
```

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

**Examples:**

Given "abcabcbb", the answer is "abc", which the length is 3.

Given "bbbb", the answer is "b", with the length of 1.

Given "pwwkew", 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.

<http://blog.csdn.net/feliciafay/article/details/16895637>

```
1. int lengthOfLongestSubstring(string s) {
2.     int n = s.length();
3.     int i = 0, j = 0;
4.     int maxLen = 0;
5.     bool exist[256] = { false };
6.     while (j < n) {
7.         if (exist[s[j]]) {
8.             maxLen = max(maxLen, j-i);
9.             while (s[i] != s[j]) {
10.                 exist[s[i]] = false;
11.                 i++;
12.             }
13.             i++;
14.             j++;
15.         } else {
16.             exist[s[j]] = true;
17.             j++;

```

```
18. }  
19. }  
20. maxlen = max(maxLen, n-i);  
21. return maxlen;  
22. }
```