

## 实现 `strStr()`

实现 `strStr()` 函数。

给定一个 `haystack` 字符串和一个 `needle` 字符串，在 `haystack` 字符串中找出 `needle` 字符串出现的第一个位置 (从 0 开始)。如果不存在，则返回 **-1**。

**示例 1:**

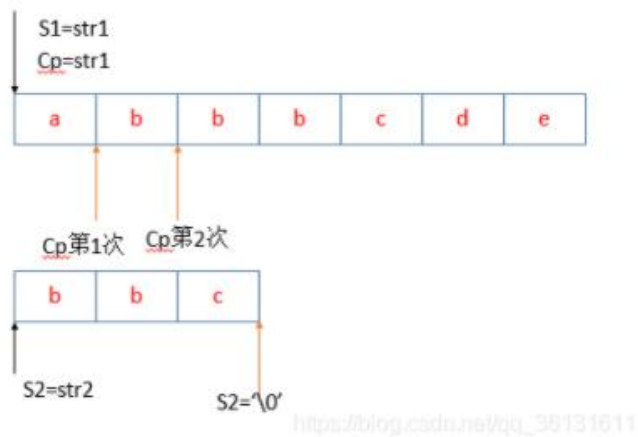
输入: `haystack = "hello"`, `needle = "ll"`

输出: 2

**示例 2:**

输入: `haystack = "aaaaa"`, `needle = "bba"`

输出: -1



模拟实现`my_strstr()`:主要难点在于一个字符一个字符逐一匹配，需要一个指针记录起始位置，每次匹配失败需要退回起始位置的下一位置处开始在次匹配

```
int strStr(char * haystack, char * needle){
    char* s1 = haystack;
    char* s2 = needle;
    char* cp = haystack;//该指针记录每一次进行匹配的起始位置
    if(!haystack||!needle)
    {
        return -1;
    }

    if (*needle == '\0')
    {

```

```
        return 0;
    }
    while (*cp)
    {
        //一次匹配查找的过程
        s1 = cp;
        s2 = needle;
        while (*s1 && *s2 && *s1 == *s2)//逐个匹配
        {
            s1++;
            s2++;
        }
        if (*s2 == '\\0')//只有 s2 走完才能说明找到真正的子串 返回起始查找位置
cp s1 走完还没有找到匹配的子串，则返回 null
        {
            return (int)(cp-haystack);
        }
        cp++;
    }
    return -1;
}
```