

## 第 1 题

某市去年全年举行某疫苗全民接种, 现今民众可以通过接种号对疫苗接种情况进行查询。接种号为 10 位, 首位大写字母表示 A,B,C 三个接种点, 2~3 位表示接种月份, 4~5 位表示疫苗的种类 (07, 13, 23), 后 5 位为接种者的数字编号。

请用**等价类划分**方法对接种号输入是否合法进行测试。要求给出**等价类表**, 和具体的**覆盖数据**。(每种无效等价类都独立给出一个覆盖数据, 并给出一个正确输入)

参考格式: [《等价类划分》-有这篇就够了 - 知乎 \(zhihu.com\)](#)

## 第 2 题

给出以下两个代码的**环路复杂度**, 并给出所有的**独立路径**, 同时对于每条独立路径给出**完整测试用例以及对应输出**。(流图中的结点标识最好能说明代表程序中的哪些代码段, 或者用**相应代码行号**作为结点标识, 若代码中存在循环, 可将**循环体视为只执行一次**, 这样方便写独立路径。测试用例不唯一, 满足路径即可, 对于**实际不存在的路径**, 进行说明后可不进行测试。)

代码一:

```
1. void lcs(int i, int j_initial, char a[], char b[])
2. {
3.     length1 = strlen(a);
4.     length2 = strlen(b);
5.     while (i <= length1)
6.     {
7.         j = j_initial;
8.         while (j <= length2)
9.         {
10.            if (a[i - 1] == b[j - 1])
11.            {
12.                arr[i][j] = arr[i - 1][j - 1] + 1; //int arr[max][max]={0}
13.            }
14.            else
15.            {
16.                if (arr[i - 1][j] > arr[i][j - 1])
17.                    arr[i][j] = arr[i - 1][j];
18.                else
19.                    arr[i][j] = arr[i][j - 1];
20.            }
21.            j++;
22.        }
23.        i++;
24.    }
25.    printf("%d\n", arr[length1][length2]);
26. }
```

代码二:

```
1. int Isprime(int n){
2.     int i;
3.     if (n < 2)
4.         printf("%d 不是素数", n);
5.     else {
6.         for (i = 2; i < n; i++) {
7.             if (n%i == 0)
8.                 break;
9.         }
10.        if (i < n) {
11.            printf("%d 不是素数\n", n);
12.        }
13.        else
14.            printf("%d 是素数\n", n);
15.    }
16.    return 0;
17. }
```

参考格式

独立路径	测试用例	结果
路径 1: 1, 2……	n = …	
路径 2: …		
路径 3: …		

第三题

请同学们安装和使用 SonarQube，对自己以前写得的程序代码进行检测。找到 5 种不同严重程度的 bug 例子并给出其相应的修复方案与修复结果。(提供截图和相应的文本描述)

SonarQube 的安装手册另见附件