

温州大学计算机与人工智能学院

程序设计基础(2023-2024-1) 课程作业

实验名称	不要62				
班 级	23大数据1	姓 名	徐王晶	学 号	23211870102
实验地点	南5B105	实验时间	2023-09-22,23:57:07	指导老师	虞铭财

一、问题编号：

1955

地址: <http://10.132.254.54/problem/1955/>

二、问题描述：

杭州人称那些傻乎乎粘嗒嗒的人为62（音：laoer）。 杭州交通管理局经常会扩充一些的士车牌照，新近出来一个好消息，以后上牌照，不再含有不吉利的数字了，这样一来，就可以消除个别的士司机和乘客的心理障碍，更安全地服务大众。 不吉利的数字为所有含有4或62的号码。例如：62315 73418 88914 都属于不吉利号码。但是，61152虽然含有6和2，但不是62连号，所以不属于不吉利数字之列。 你的任务是，对于每次给出的一个牌照区间号，推断出交管局今次又要实际上给多少辆新的士车上牌照了。

三、输入说明：

输入的都是整数对m、n（ $0 < m \leq n \leq 1000$ ），如果遇到都是0的整数对，则输入结束。

四、输出说明：

对于每个整数对，输出吉利数字的总数，该数值占一行位置。

五、输入样列：

```
1 100
0 0
```

六、输出样列：

```
80
```

七、解答内容：

所用语言：

源代码：

```
01. #include <stdio.h>
02.
03. int fun(int a);
04.
05. int main(void)
06. {
07.     int m, n;
08.     int s;
09.
10.     scanf("%d %d", &m, &n);
11.
12.     while ((m != 0) && (n != 0))
13.     {
14.         s = n - m + 1;
15.
16.         for (; m <= n; m++)
17.         {
18.             if (!fun(m))
19.             {
20.                 s -= 1;
21.             }
22.         }
23.
24.         printf("%d\n", s);
25.         scanf("%d %d", &m, &n);
26.     }
27.
28.
29.     return 0;
30. }
31.
32. int fun(int a)
33. {
34.     int lst[1000];
35.     int p = 0;
36.     int x, y;
37.
38.     x = a / 10;
39.     y = a % 10;
40.
41.     while (x >= 1)
42.     {
43.         lst[p] = y;
44.         y = x % 10;
45.         x /= 10;
46.
47.         p++;
48.     }
49.     lst[p] = y;
50.
51.     int flag = 0;
52.     for (; p >= 0; p--)
53.     {
54.         int c = lst[p];
55.
56.         if (c == 4)
57.         {
58.             return 0;
59.         }
60.         else if (c == 6)
61.         {
62.             flag = 1;
63.         }
64.         else if ((c == 2) && (flag))
65.         {
66.             return 0;
67.         }
68.         else if (flag)
69.         {
70.             flag = 0;
71.         }
72.     }
73.
74.     return 1;
75. }
```

八、判题结果

AC - 答案正确

判题结果补充说明：

test id:3740,result:AC, usedtime:0MS, usedmem:812KB,score:50 test id:3741,result:AC, usedtime:0MS,
usedmem:812KB,score:50