

# 温州大学计算机与人工智能学院

## 程序设计课程设计 实验报告

实验名称	银行排队之一：入队				
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	2023-12-26,16:49:39	指导老师	虞铭财

### 一、问题编号：

1165

地址：<http://10.132.254.54/problem/1165/>

### 二、问题描述：

注意：用链表完成

现在银行都有排队叫号系统，如果你到银行去办理业务，首先取得一个顺序号，并告诉你前面有多少人在等待。

现在请你设计、并实现该系统。

客户信息包括：取号的号码、客户银行卡号码、客户类型（包括普通客户和vip客户，分别用Ordinary和VIP表示）。

本题要求实现以下功能：

（1）客户进入银行并取号，该功能对应的命令为IN number type，number表示客户银行卡号码，type表示客户类型。

（2）退出系统，该功能对应的命令为QUIT。

### 三、输入说明：

输入有多行，每行表示一种操作。

本题实现的操作有：

IN guestnumber guesttype

其中IN 表示入队，guestnumber表示客户银行卡号码，guesttype有两种取值，分别为Ordinary和VIP，表示普通客户与VIP客户。

QUIT

退出银行排队系统

请注意：输入的数据中，只有最后一个命令是QUIT命令。

### 四、输出说明：

输出有多行，对不同的命令给出不同的输出。

以下是具体的输出说明：

IN 客户银行卡号码 客户类型

该命令首先在单独的一行中输出"IN:"，然后再输出客户信息。

客户信息包括客户编号 客户银行卡号码 客户类型 该客户前面的客户数。客户编号由系统在客户取号的时候给定，从1开始顺序编号。

客户类型分为Ordinary和VIP。数据之间用一个空格分开。

请注意：

第一个客户入队后，则该用户的顺序号为1，前面有0个客户等待；

第二个客户入队后，则该用户的顺序号为2，前面有1个客户等待。

以此类推。

QUIT

在单独的一行中显示"GOOD BYE!"后结束程序。

### 五、输入样例：

IN 1000001 Ordinary

IN 2000003 VIP

IN 2000009 VIP

QUIT

六、输出样例：

```
IN:1 1000001 Ordinary 0
IN:2 2000003 VIP 1
IN:3 2000009 VIP 2
GOOD BYE!
```

## 七、解答内容：

所用语言：

源代码：

```
01. #include <stdio.h>
02. #include <stdlib.h>
03. #include <string.h>
04.
05. struct Data
06. {
07.     char id[20];
08.     char cla[10];
09.     int num;
10.     struct Data *prev;
11. };
12.
13. struct Queue
14. {
15.     struct Data *head;
16.     struct Data *tail;
17. };
18.
19. void In(struct Queue *queue);
20.
21. int main(void)
22. {
23.     struct Queue queue;
24.     queue.head = NULL;
25.     queue.tail = NULL;
26.
27.     char op[20];
28.     while (scanf("%s", op), strcmp(op, "QUIT") != 0)
29.     {
30.         if (strcmp(op, "IN") == 0)
31.         {
32.             In(&queue);
33.         }
34.     }
35.     printf("GOOD BYE!\n");
36.     return 0;
37. }
38.
39. void In(struct Queue *queue)
40. {
41.     struct Data *data = (struct Data *)malloc(sizeof(struct Data));
42.     scanf(
43.         "%s %s",
44.         data -> id,
45.         data -> cla
46.     );
47.
48.     if (queue -> head == NULL)
49.     {
50.         queue -> head = data;
51.         queue -> tail = data;
52.         data -> prev = NULL;
53.         data -> num = 0;
54.     }
55.     else
56.     {
57.         data -> prev = queue -> tail;
58.         data -> num = queue -> tail -> num + 1;
59.         queue -> tail = data;
60.     }
61.
62.     printf(
63.         "IN:%d %s %s %d\n",
64.         queue -> tail -> num + 1,
65.         queue -> tail -> id,
66.         queue -> tail -> cla,
67.         queue -> tail -> num
68.     );
69.
70. }
```

## 八、判题结果

**AC - 答案正确**

判题结果补充说明：

test id:2027,result:AC, usedtime:0MS, usedmem:864KB,score:100