1. 代码构思

```
事件分为四种类型:0—事件到达，1—事件从队伍1离开，2—从队伍2离开，3—从队伍1切换到2
凡是离开都要renew total,totaltime,进入customer++;
0—插入事件(en.OccurTime+interval<closetime)：状态0，time=en.Occur+interval
 —新到customer插入wait1
 —if(wait1.length==1&&(QueueLength(wait2)==0?1:(total+GetHead(wait2)->bill<0)))
     —if:total+GetHead(wait1)->bill>=0:
         插入事件(wait1.length=1)：当前的从队伍1离开—状态=1，time=en.Occur+qnew->duration
     —else:total+GetHead(wait1)->bill<0:
         插入事件：状态=3,time=en.Occur;
//1这里有必然触发的事件吗
1—if(wait2!=0&&total+wait2.head->bill>=0)
         插入事件：状态=2，time=en.occur+wait2.head->duration
   else  if(wait1.length!=0):
             if(total+gethead(wait1)->bill>=0)
                 插入事件：状态=1，time=en.occur+gethead(wait1)->duration
             else 插入事件：状态=3，time=en.occur
  else 啥都干不了啦
2—if(wait2!=0&&total+wait2.head->bill>=0)   //wait2还能接着出先出wait2
         插入事件：状态=2，time=en.occur+wait2.head->duration
   else if(wait1.length!=0):     //wait1不空的话能出出，不能移走
             if(total+gethead(wait1)->bill>=0)
                 插入事件：状态=1，time=en.occur+gethead(wait1)->duration
             else 插入事件：状态=3，time=en.occur
3—if(wait1.length!=0):  //wait1不空的话能出出，不能移走
             if(total+gethead(wait1)->bill>=0)
                 插入事件：状态=1，time=en.occur+gethead(wait1)->duration
             else 插入事件：状态=3，time=en.occur
```

2.艰难debug

1. rand生成一样的bill和interval

- srand()写在循环外面

2. customer=0

☑ 维护：insert\del\init 都要改变head/tail

3. 到达即离开

☑ duration=rand()%10+1

4. 计算人均用时

- 结束时，对wait2中的每个队伍wait1中的成员a最后都会分到wait2，total+=closetime-a->arrivtime;

5. QueueLength(wait2)异常，

- whyQueueLength(wait1)没问题，
- a=0xFFFFFFFFFFF7还会进行a!=NULL判断，说明rear->next!=NULL;没分配?

```
//将wait1首转入wait2,
qnew=GetHead(wait1);
DelQueue(wait1);    //这步作死把qnew释放了，qnew变为野指针,后面QueueLength(wait2)赋给a,会出现
a=0xF..F7
EnQueue(wait2,qnew);
```

- DelQueu()定义中杠掉free(fr)即可

6. 一个老问题——循环中用链表长度做边界，而这个边界是会随着链表的删减变动的呀！

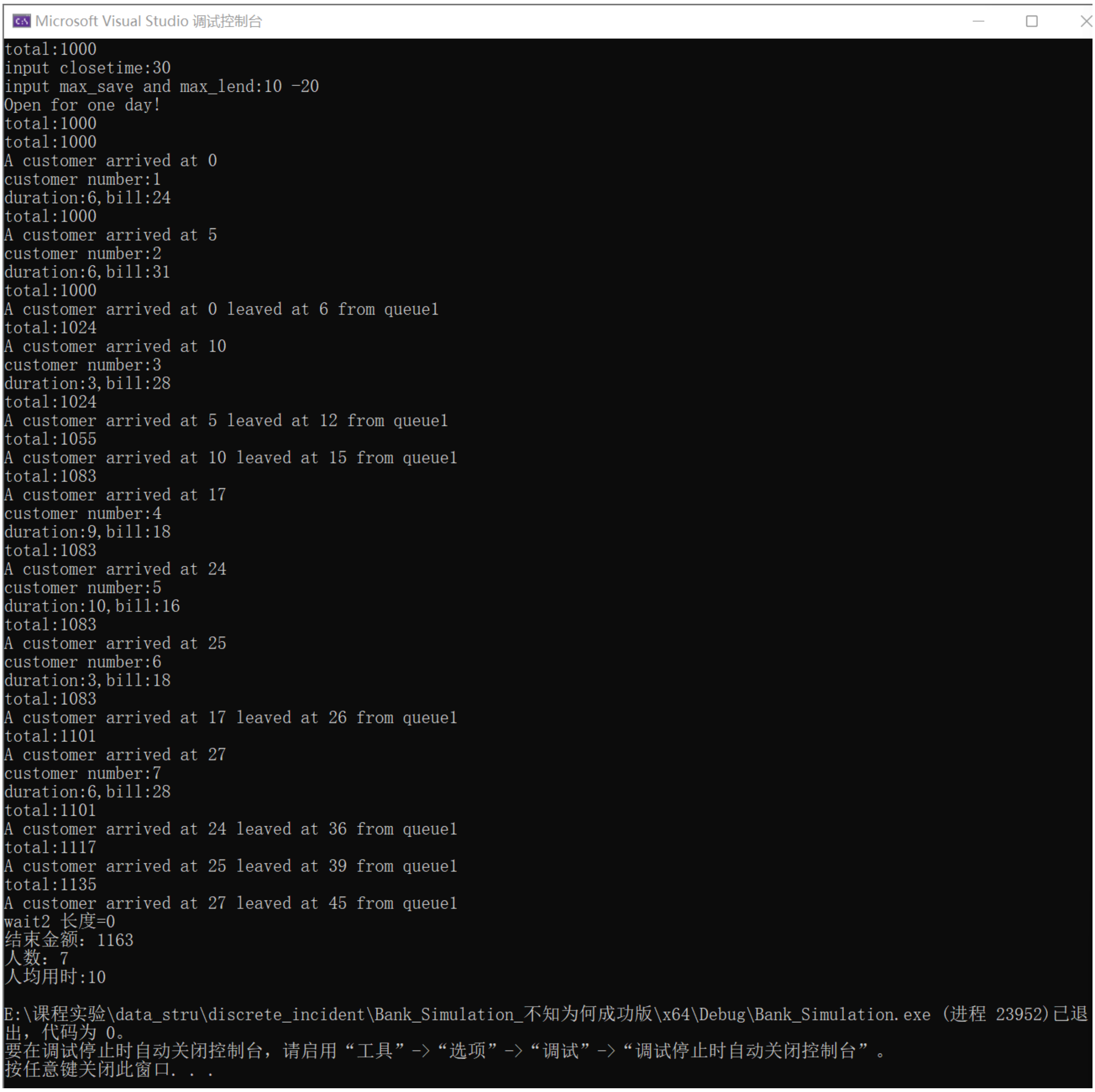
```
    for (int i = 0; i < QueueuLength(wait2); i++) {
        Totaltime += closetime - GetHead(wait2)->Arrivtime;
        DelQueue(wait2);
        printf("这是第%d次\n", i);
    }
//解决方案：for循环前把QueueLength(wait2)赋给变量，后作为边界
```

3.代码解释

4.运行结果（从能跑开始）

只有借款

```
下面这个没细想,mkQNode函数中令
mk->bill=rand()%(max_save-max_lend)+(max_save-max_lend)/2,
但可以说明入队1没问题
实际应是
mk->bill=rand()%(max_save-max_lend)+max_lend,才有借款取款,更正后的运行结果在此之后
且为了能出现从1移到2，total初始化=0;
```



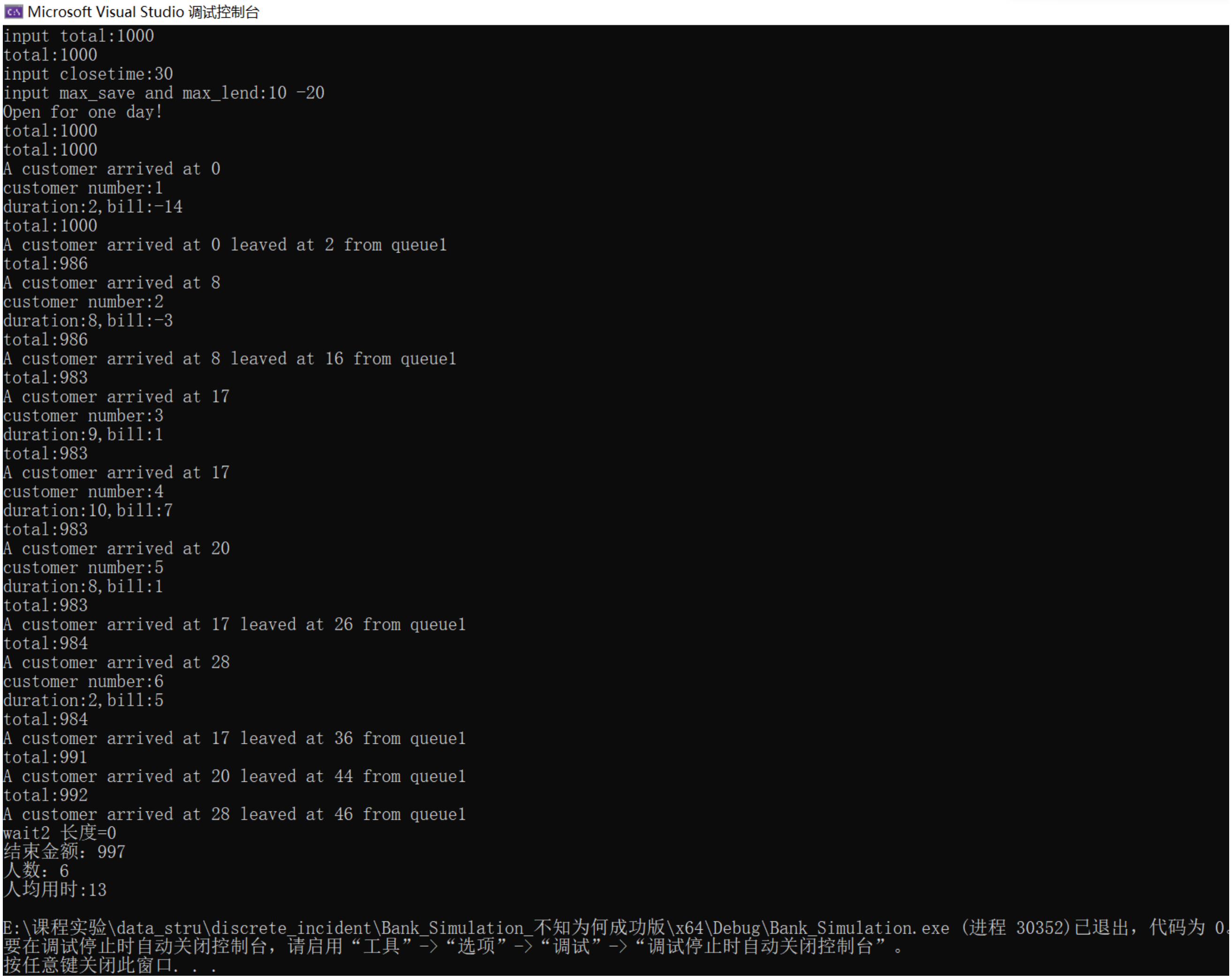
```
total:1000
input closetime:30
input max_save and max_lend:10 -20
Open for one day!
total:1000
total:1000
A customer arrived at 0
customer number:1
duration:6,bill:24
total:1000
A customer arrived at 5
customer number:2
duration:6,bill:31
total:1000
A customer arrived at 0 leaved at 6 from queue1
total:1024
A customer arrived at 10
customer number:3
duration:3,bill:28
total:1024
A customer arrived at 5 leaved at 12 from queue1
total:1055
A customer arrived at 10 leaved at 15 from queue1
total:1083
A customer arrived at 17
customer number:4
duration:9,bill:18
total:1083
A customer arrived at 24
customer number:5
duration:10,bill:16
total:1083
A customer arrived at 25
customer number:6
duration:3,bill:18
total:1083
A customer arrived at 17 leaved at 26 from queue1
total:1101
A customer arrived at 27
customer number:7
duration:6,bill:28
total:1101
A customer arrived at 24 leaved at 36 from queue1
total:1117
A customer arrived at 25 leaved at 39 from queue1
total:1135
A customer arrived at 27 leaved at 45 from queue1
wait2 长度=0
结束金额: 1163
人数: 7
人均用时:10

E:\课程实验\data_stru\discrete_incident\Bank_Simulation_不知为何成功版\x64\Debug\Bank_Simulation.exe (进程 23952)已退
出，代码为 0。
要在调试停止时自动关闭控制台，请启用“工具”->“选项”->“调试”->“调试停止时自动关闭控制台”。
按任意键关闭此窗口. . .
```

time	customer_number	total	totaltime
0	1(1号，daration=6,bill=24)	1000	0
5	2(2号duration=6,bill=31)	1000	0
6	2(1号离开)	1000+24=1024	0+(6-0)=6
10	3(3号，duration=3,bill=28)	1024	6
12	3(2号离开)	1024+31=1055	6+(12-5)=13
15	3(3号离开)	1055+28=1083	13+(15-10)=18
17	4(4号，duration=9,bill=18)	1083	18
24	5(5号，duration=10,bill=16)	1083	18
25	6(6号，duration=3,bill=18)	1083	18
26	6(4号离开)	1083+18=1101	18+(26-17)=27
27	7(7号，duration=6,bill=28)	1101	27
36	7(5号离开)	1101+16=1117	27+(36-24)=39
39	7(6号离开)	1117+18=1135	39+(39-25)=53
45	7(7号离开)	1135+28=1163	53+(45-27)=71
总	7	1163	71；平均用时71/10=7

更正后，

借款+还款

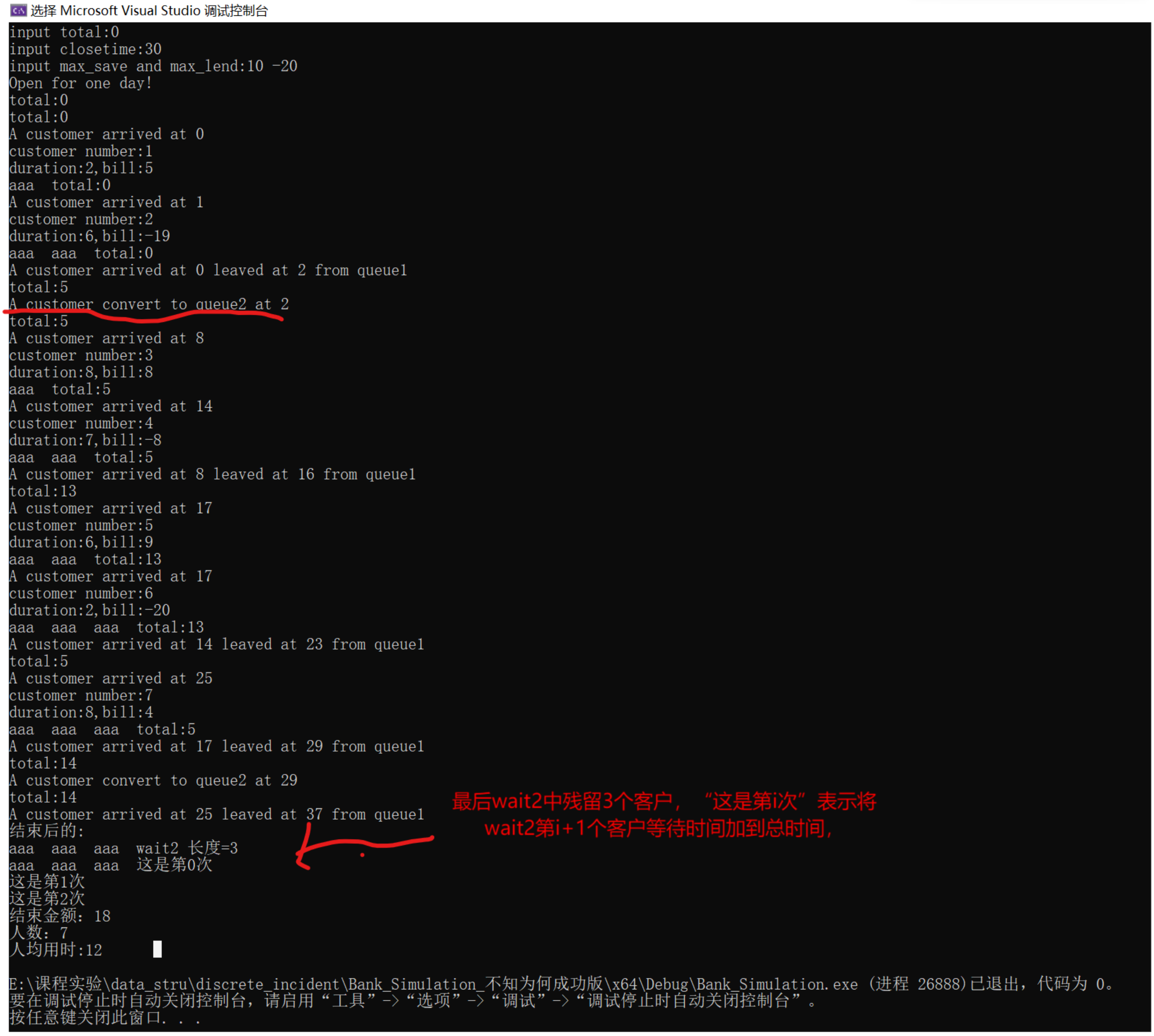
但是因为上面的问题5、6换队操作QueueLength(wait2)会出问题



```
input total:1000
total:1000
input closetime:30
input max_save and max_lend:10 -20
Open for one day!
total:1000
total:1000
A customer arrived at 0
customer number:1
duration:2,bill:-14
total:1000
A customer arrived at 0 leaved at 2 from queue1
total:986
A customer arrived at 8
customer number:2
duration:8,bill:-3
total:986
A customer arrived at 8 leaved at 16 from queue1
total:983
A customer arrived at 17
customer number:3
duration:9,bill:1
total:983
A customer arrived at 17
customer number:4
duration:10,bill:7
total:983
A customer arrived at 20
customer number:5
duration:8,bill:1
total:983
A customer arrived at 17 leaved at 26 from queue1
total:984
A customer arrived at 28
customer number:6
duration:2,bill:5
total:984
A customer arrived at 17 leaved at 36 from queue1
total:991
A customer arrived at 20 leaved at 44 from queue1
total:992
A customer arrived at 28 leaved at 46 from queue1
wait2 长度=0
结束金额：997
人数：6
人均用时:13

E:\课程实验\data_stru\discrete_incident\Bank_Simulation_不知为何成功版\x64\Debug\Bank_Simulation.exe (进程 30352)已退出，代码为 0。
要在调试停止时自动关闭控制台，请启用“工具”->“选项”->“调试”->“调试停止时自动关闭控制台”。
按任意键关闭此窗口. . .
```

借款+还款+换队+(从二队出队失败)

解决5.6后，可计算答案的正确性，

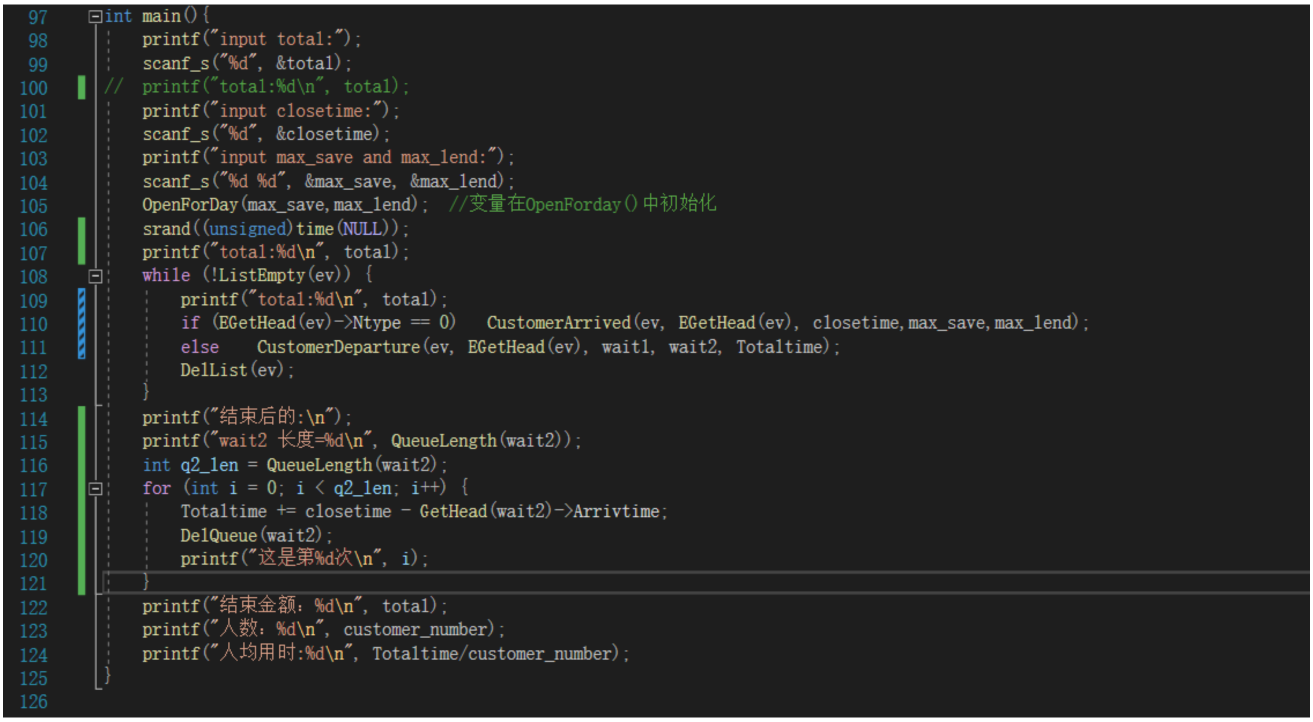


```
input total:0
input closetime:30
input max_save and max_lend:10 -20
Open for one day!
total:0
total:0
A customer arrived at 0
customer number:1
duration:2,bill:5
aaa  total:0
A customer arrived at 1
customer number:2
duration:6,bill:-19
aaa  aaa  total:0
A customer arrived at 0 leaved at 2 from queue1
total:5
A customer convert to queue2 at 2
total:5
A customer arrived at 8
customer number:3
duration:8,bill:8
aaa  total:5
A customer arrived at 14
customer number:4
duration:7,bill:-8
aaa  aaa  total:5
A customer arrived at 8 leaved at 16 from queue1
total:13
A customer arrived at 17
customer number:5
duration:6,bill:9
aaa  aaa  total:13
A customer arrived at 17
customer number:6
duration:2,bill:-20
aaa  aaa  aaa  total:13
A customer arrived at 14 leaved at 23 from queue1
total:5
A customer arrived at 25
customer number:7
duration:8,bill:4
aaa  aaa  aaa  total:5
A customer arrived at 17 leaved at 29 from queue1
total:14
A customer convert to queue2 at 29
total:14
A customer arrived at 25 leaved at 37 from queue1
结束后的:
aaa  aaa  aaa  wait2 长度=3
aaa  aaa  aaa  这是第0次
这是第1次
这是第2次
结束金额: 18
人数: 7
人均用时:12

E:\课程实验\data_stru\discrete_incident\Bank_Simulation_不知为何成功版\x64\Debug\Bank_Simulation.exe (进程 26888)已退出，代码为 0。
要在调试停止时自动关闭控制台，请启用 “工具” -> “选项” -> “调试” -> “调试停止时自动关闭控制台” 。
按任意键关闭此窗口. . .
```

最后wait2中残留3个客户， “这是第i次”表示将wait2第i+1个客户等待时间加到总时间，

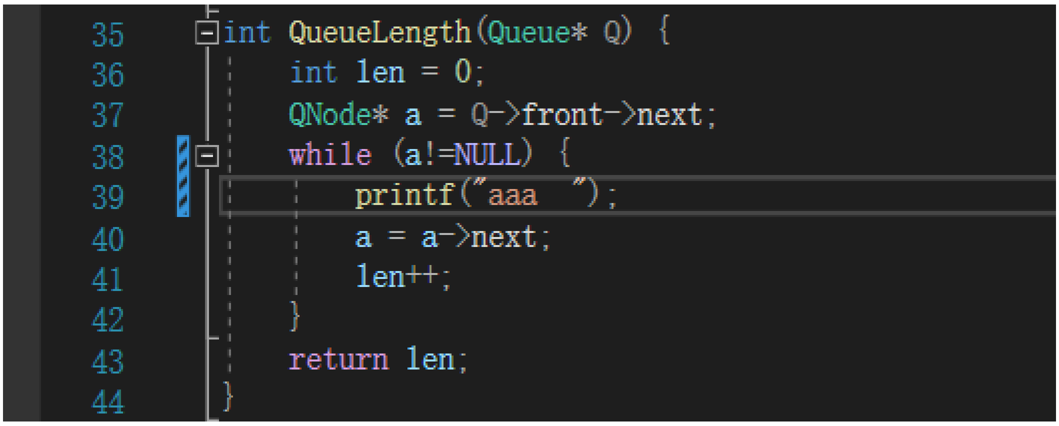
末尾的aaa，wait2长度，这是第i次是debug所用，但这个例子比较难出，就不更了

相关代码：



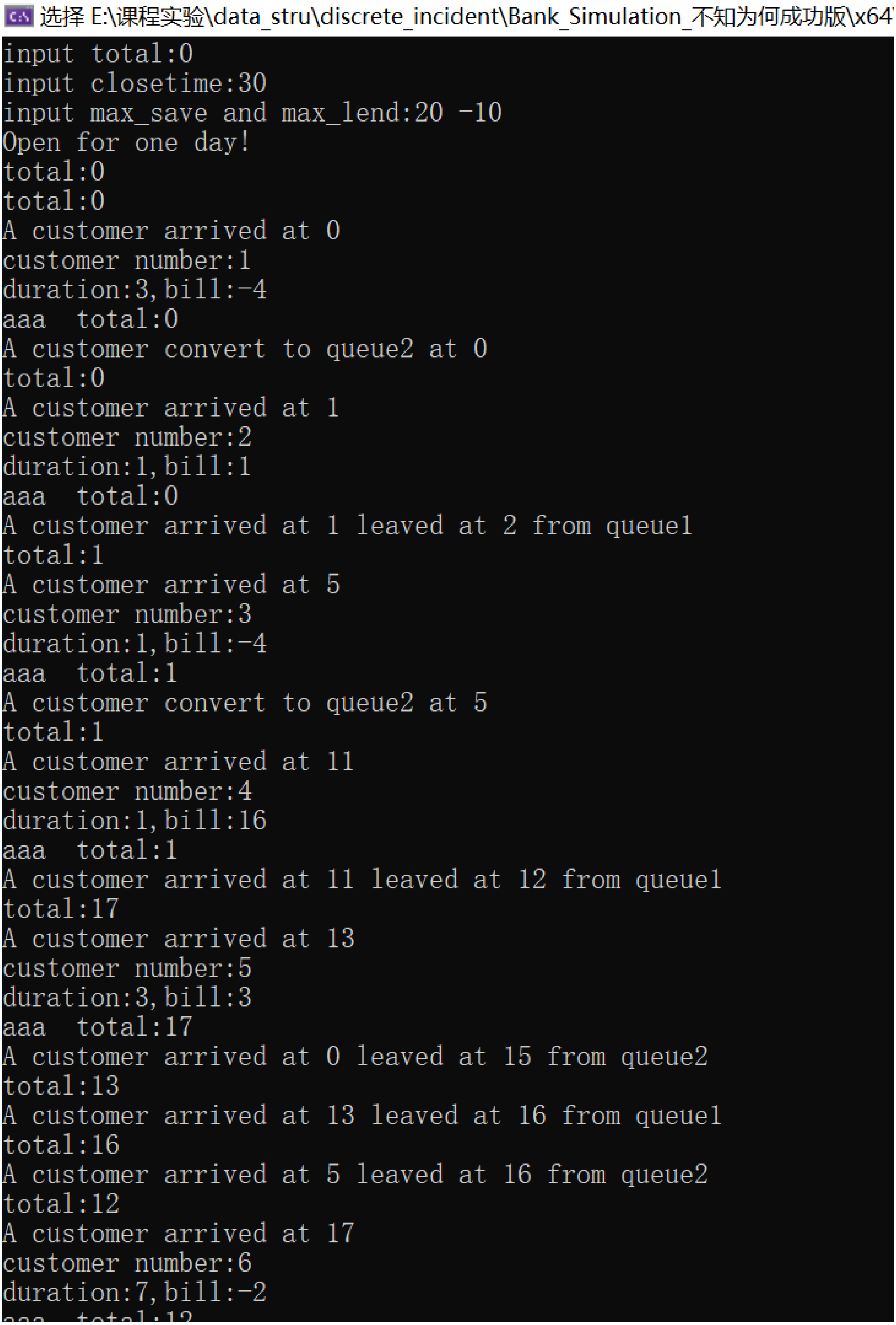
```
97   int main(){
98       printf("input total:");
99       scanf_s("%d", &total);
100  //  printf("total:%d\n", total);
101      printf("input closetime:");
102      scanf_s("%d", &closetime);
103      printf("input max_save and max_lend:");
104      scanf_s("%d %d", &max_save, &max_lend);
105      OpenForDay(max_save,max_lend);  //变量在OpenForday()中初始化
106      srand((unsigned)time(NULL));
107      printf("total:%d\n", total);
108      while (!ListEmpty(ev)) {
109          printf("total:%d\n", total);
110          if (EGetHead(ev)->Ntype == 0)   CustomerArrived(ev, EGetHead(ev), closetime,max_save,max_lend);
111          else    CustomerDeparture(ev, EGetHead(ev), wait1, wait2, Totaltime);
112          DelList(ev);
113      }
114      printf("结束后的:\n");
115      printf("wait2 长度=%d\n", QueueLength(wait2));
116      int q2_len = QueueLength(wait2);
117      for (int i = 0; i < q2_len; i++) {
118          Totaltime += closetime - GetHead(wait2)->Arrivtime;
119          DelQueue(wait2);
120          printf("这是第%d次\n", i);
121      }
122      printf("结束金额：%d\n", total);
123      printf("人数：%d\n", customer_number);
124      printf("人均用时:%d\n", Totaltime/customer_number);
125  }
126
```

115行调用了如下QueueLength()，才会出现aaa aaa aaa



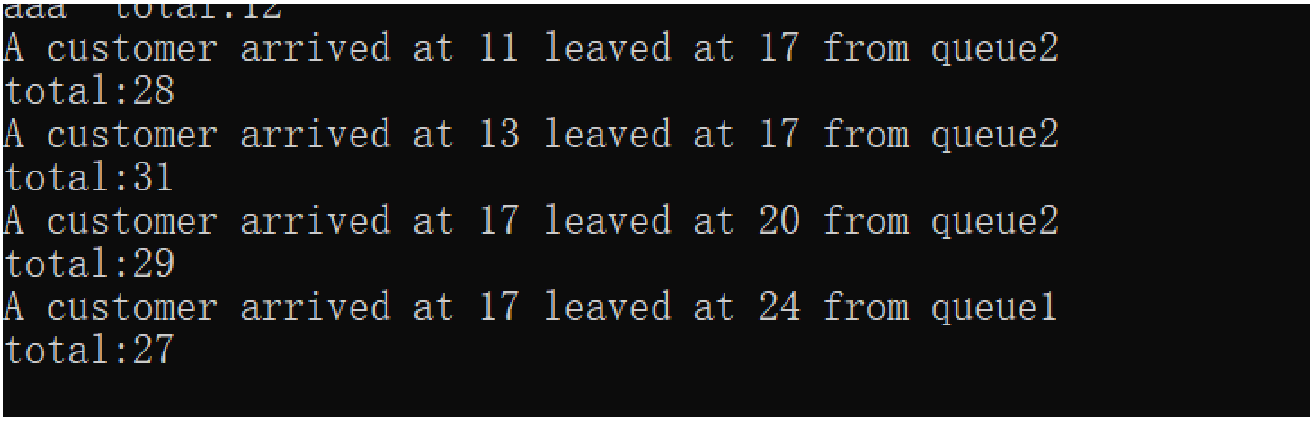
```
int QueueLength(Queue* Q) {
    int len = 0;
    QNode* a = Q->front->next;
    while (a!=NULL) {
        printf("aaa  ");
        a = a->next;
        len++;
    }
    return len;
}
```

可从二队出队版

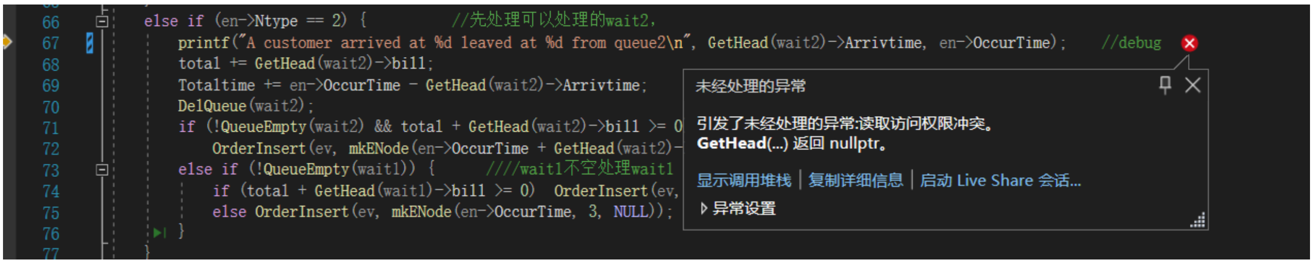


选择 E:\课程实验\data_stru\discrete_incident\Bank_Simulation_不知为何成功版\x64

```
input total:0
input closetime:30
input max_save and max_lend:20 -10
Open for one day!
total:0
total:0
A customer arrived at 0
customer number:1
duration:3,bill:-4
aaa  total:0
A customer convert to queue2 at 0
total:0
A customer arrived at 1
customer number:2
duration:1,bill:1
aaa  total:0
A customer arrived at 1 leaved at 2 from queue1
total:1
A customer arrived at 5
customer number:3
duration:1,bill:-4
aaa  total:1
A customer convert to queue2 at 5
total:1
A customer arrived at 11
customer number:4
duration:1,bill:16
aaa  total:1
A customer arrived at 11 leaved at 12 from queue1
total:17
A customer arrived at 13
customer number:5
duration:3,bill:3
aaa  total:17
A customer arrived at 0 leaved at 15 from queue2
total:13
A customer arrived at 13 leaved at 16 from queue1
total:16
A customer arrived at 5 leaved at 16 from queue2
total:12
A customer arrived at 17
customer number:6
duration:7,bill:-2
aaa  total:12
```



```
aaa  total:12
A customer arrived at 11 leaved at 17 from queue2
total:28
A customer arrived at 13 leaved at 17 from queue2
total:31
A customer arrived at 17 leaved at 20 from queue2
total:29
A customer arrived at 17 leaved at 24 from queue1
total:27
```



```
else if (en->Ntype == 2) {          //先处理可以处理的wait2,
    printf("A customer arrived at %d leaved at %d from queue2\n", GetHead(wait2)->Arrivtime, en->OccurTime);    //debug
    total += GetHead(wait2)->bill;
    Totaltime += en->OccurTime - GetHead(wait2)->Arrivtime;
    DelQueue(wait2);
    if (!QueueEmpty(wait2) && total + GetHead(wait2)->bill >= 0
        OrderInsert(ev, mkENode(en->OccurTime + GetHead(wait2)-
    else if (!QueueEmpty(wait1)) {       ////wait1不空处理wait1
        if (total + GetHead(wait1)->bill >= 0)  OrderInsert(ev,
        else OrderInsert(ev, mkENode(en->OccurTime, 3, NULL));
    }
}
```

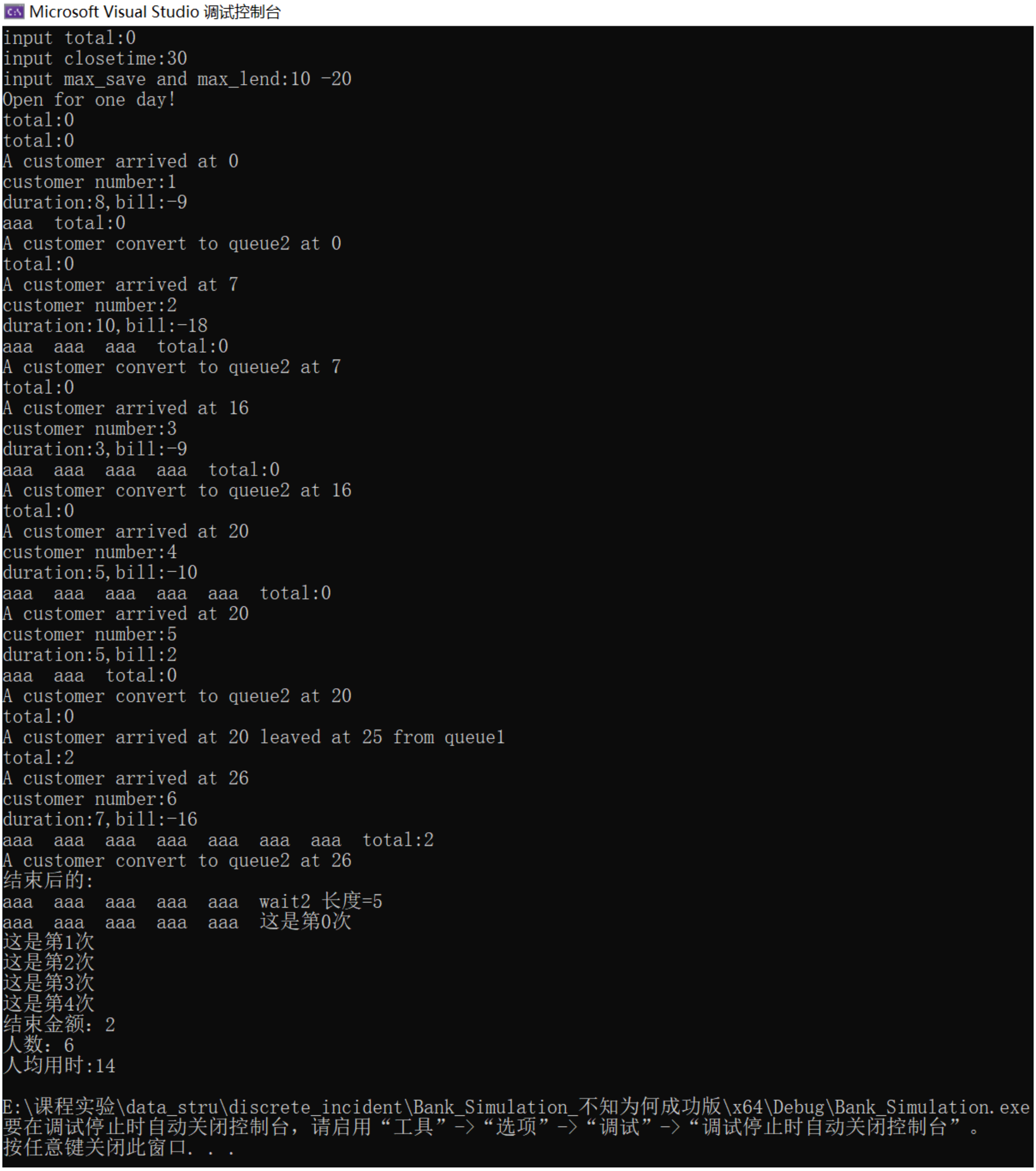
未经处理的异常

引发了未经处理的异常:读取访问权限冲突。
GetHead(...) 返回 nullptr。

上面在5和13到达的同时在16离开，是因为4号离开后，wait1=0,wait2=2,在wati2出队的时候错误地让这期间到达的5号也出队，修改到达即离开的限定条件为

```
QueueLength(wait1)==1&&(QueueLength(wait2)==0?1:(total+GetHead(wait2)->bill<0)
```

规范wait1到达即出队条件后版



Microsoft Visual Studio 调试控制台

```
input total:0
input closetime:30
input max_save and max_lend:10 -20
Open for one day!
total:0
total:0
A customer arrived at 0
customer number:1
duration:8,bill:-9
aaa  total:0
A customer convert to queue2 at 0
total:0
A customer arrived at 7
customer number:2
duration:10,bill:-18
aaa  aaa  aaa  total:0
A customer convert to queue2 at 7
total:0
A customer arrived at 16
customer number:3
duration:3,bill:-9
aaa  aaa  aaa  aaa  total:0
A customer convert to queue2 at 16
total:0
A customer arrived at 20
customer number:4
duration:5,bill:-10
aaa  aaa  aaa  aaa  aaa  total:0
A customer arrived at 20
customer number:5
duration:5,bill:2
aaa  aaa  total:0
A customer convert to queue2 at 20
total:0
A customer arrived at 20 leaved at 25 from queue1
total:2
A customer arrived at 26
customer number:6
duration:7,bill:-16
aaa  aaa  aaa  aaa  aaa  aaa  aaa  total:2
A customer convert to queue2 at 26
结束后的:
aaa  aaa  aaa  aaa  aaa  wait2 长度=5
aaa  aaa  aaa  aaa  aaa  这是第0次
这是第1次
这是第2次
这是第3次
这是第4次
结束金额：2
人数：6
人均用时:14

E:\课程实验\data_stru\discrete_incident\Bank_Simulation_不知为何成功版\x64\Debug\Bank_Simulation.exe
要在调试停止时自动关闭控制台，请启用“工具”->“选项”->“调试”->“调试停止时自动关闭控制台”。
按任意键关闭此窗口. . .
```