# 银行模拟

朱河勤 PB16030899

2017 年 11 月 20 日

# 目录

# 1 需求分析

**1** 创建银行排队窗口，还可以增加vip通道，可通过队列来实现，队列可以用链表封装得到

**2** 每个客户可以存款，取款，如果银行钱不够，让客户处于等待状态，直到银行的钱足够

**3** 如果排队的队伍过长，在最后的可以选择较短的队伍

**4** 把所有事件打印出来

# 2 概要设计

## 2.1 链表

ADT linkList{

**数据对象** ：D=$\{a_i|a_i, i = 0, 12, \ldots, n\}$

**数据关系** ：R1 $= \{< a_{i-1}, a_i > |a_{i-1}, a_i in D, i = 2, 3, \ldots, n\}$

**基本操作** ：

**s = linkList()** 创建一个链表s

**s.append(x)** 向lst尾部增加节点x

**x = s[i]** 将x引用链表第i个节点

**del s[i]** 删除链表第i个节点

**s[i] = x** 将链表第i个节点赋值x

**s.insert(i,x)** 将链表第i个节点前插入x

## 2.2 双端队列

ADT linkList{

**数据对象** ：D=$\{a_i|a_i, i = 0, 12, \ldots, n\}$

**数据关系** ：R1 $= \{a_i, <a_{i-1}>|a_{i-1}, a_i inD, i = 2, 3, \ldots, n\}$

**基本操作** ：

**s = queue()** 创建一个队列s

**s.append(x)** 向lst尾部增加节点x

**x = s[i]** 将x引用第i个节点

**s.pop()** 右边第一个元素出队

**s.popleft()** 左边第一个元素出队

**s.appendleft()** 左边第一个元素进队 }

# 3　详细设计

## 3.1　用户类型

```
class client(builtins.object)
 |  Methods defined here:
 |
 |  __init__(self, serviceTime, money)
 |      客户数据：包括客户数，总等待时间，用户名，离开的时间
 |      到达的时间，服务需求的时间，存款取款的金额
 |
 |  ----------------------------------------------------------------------
 |  Data descriptors defined here:
 |
 |  __dict__
 |      dictionary for instance variables (if defined)
 |
 |  __weakref__
 |      list of weak references to the object (if defined)
 |
 |  ----------------------------------------------------------------------
 |  Data and other attributes defined here:
 |
 |  nameMaxLength = 0
 |
 |  names = ['Jacob', 'Michael', 'Ethan', 'Joshua', 'Alexander', 'Anthony'...
 |
 |  num = 0
 |
 |  waitTime = 0
```

## 3.2 窗口类型

```
class window(builtins.object)
 |  窗口类型，通过封装，可以当作双端队列使用，而且也支持随机访问（但是是O(n)时间）
 |
 |  Methods defined here:
 |
 |  __add__(self, s)
 |      加法
 |
 |  __bool__(self)
 |      判断是否为空
 |
 |  __delitem__(self, x)
 |      删除第i个元素
 |
 |  __getitem__(self, i)
 |      取得第i个元素
 |
 |  __iadd__(self, s)
 |      原地加法
 |
 |  __init__(self, no=0, s=None)
 |      initialization
 |
 |  __len__(self)
 |      求队伍长度
 |
 |  __repr__(self)
 |      Return repr(self).
 |
 |  __setitem__(self, i, x)
 |      赋值第i个元素
 |
 |  __str__(self)
 |      Return str(self).
 |
 |  append(self, x)
 |      右边尾部插入一个元素
 |
 |  appendleft(self, x)
 |      左边头部插入一个元素
 |
 |  pop(self, i=-1)
 |      右边第一个元素出队
 |
 |  popleft(self)
 |      左边第一个元素出队
```

## 3.3　银行类型

```
class bank(builtins.object)
 |  Methods defined here:
 |
 |  __init__(self, totalMoney=10000, openTime=100, serviceTime=20, money=500, n=2)
 |      银行数据结构包括：自定义个数的窗口，vip窗口，临时等待窗口银行总金额，服务时间，以及记录当前时间
 |
 |  __len__(self)
 |      返回窗口个数
 |
 |  addInfo(self, window)
 |
 |  arrange(self)
 |      调整队伍，使较长的队伍尾部转到较短的
 |
 |  checkWaitWindow(self)
 |      当银行有收入时，检查能否满足临时等待窗口的客户的取款金额
 |
 |  clientLeave(self, window)
 |      客户离开，处理金额交易
 |
 |  close(self)
 |      结束
 |
 |  display(self)
 |      即时显示事件，窗口客户等所以信息
 |
 |  goOn(self)
 |      总的控制函数
 |
 |  newClient(self)
 |      随机产生一个客户，随机设置客户信息
 |
 |  onService(self, idx)
 |      处理每个窗口是否正在服务
 |
 |  open(self)
 |      开始
 |
 |  scrollInfo(self)
 |      管理所有事件信息
 |
 |  ----------------------------------------------------------------------
```

## 3.4 显示信息的函数代码

```python
def display(self):
    '''即时显示事件，窗口客户等所以信息'''
    os.system('cls')
    print('Time  %d/%d \t bankMoney $%d'%(bank.curTime,self.openTime,self.totalMoney))
    for idx,i in enumerate(self.windows):
        s = []
        Time = []
        for j in i:
            Time.append(str(bank.curTime-j.arrive))
            s.append(j.name)
        print('window %d has %d people:'%(idx+1,len(i)),end=' ')
        s = [client.uniLengthNames[i] for i in s]
        print(''.join(s))
        print('    total waitTime      ',end=' ')
        Time = [i+(client.nameMaxLength-len(i))*' ' for i in Time]
        if Time:Time[0] = ' '*client.nameMaxLength
        print(''.join(Time))
    print('')
    wait = [client.uniLengthNames[i.name] for i in self.waitWindow]
    print('waitting window :'+''.join(wait))
    self.scrollInfo()
def scrollInfo(self):
    '''管理所有事件信息'''
    print('')
    print('*'*100)
    #if len(self.scroll)>10:self.scroll=self.scroll[-10:]
    for i,info in enumerate(self.scroll):
        '''if info[0] == ' ':
            info = info[2:]
        else:
            info = ' '*20+ info'''
        print('#{:<3d}'.format(i+1)+info)
    print('*'*100)
    print('')
```

# 4　调试分析

**1**　整体上，定义了window，client，bank类型，主要的操作在bank类中的方法实现

**2**　为了标识每个用户，我给每个用户设置了一个英文名（可以有重复）。我在网上搜索了最常用的100个英文名，然后赋值到python交互模式，以字符串保存，然后用正则表达式匹配英文名

**3**　在输出信息上，我通过清屏来更新信息。为了对齐更加美观，为另存了一个字典来保存每个客户统一长度的名字

# 5 用户手册

本程序需要在安装了python下的环境运行，可以在python的ide中使用文件运行，也可以通过命令行运行python bankSimaulate.py

下面为程序运行界面

# 6 测试结果

## 6.1 输入10000，100，20，50，2

```
Time   100/100    bankMoney $442
window 1 has 10 people: Diego    Andrew    Gabriel   Elijah    Maria     Aaliyah   Isaiah    Kimberly   Anna
      total waitTime            40        36        33        30        23        19        16         13
window 2 has 10 people: Chase    Mia       Kaden     Olivia    Bailey    Cole      Jaden     Mackenzie  Megan
      total waitTime            42        35        32        28        21        17        14         8

waitting window :Brian

********************************************************************************************
#1   Jessica has deposited $432.75 @ Time 4 @ window 1
#2   Angel has withdrew $129.35 @ Time 5 @ window 2
#3   VIP Paige gets service quickly
#4   Juan has withdrew $97.55 @ Time 16 @ window 2
#5   Ayden has deposited $70.54 @ Time 19 @ window 1
#6   Mary has withdrew $429.96 @ Time 24 @ window 1
#7   Genesis has withdrew $15.04 @ Time 31 @ window 2
#8   Jocelyn has withdrew $358.59 @ Time 33 @ window 1
#9   Abigail has deposited $51.6 @ Time 39 @ window 2
#10  Brayden has withdrew $342.1 @ Time 40 @ window 2
#11  Audrey has deposited $378.33 @ Time 45 @ window 2
#12  Isabelle has deposited $232.13 @ Time 49 @ window 1
#13  what a pity! Amelia has to wait because the poor bank has not enough money to withdraw
#14  what a pity! Eli has to wait because the poor bank has not enough money to withdraw
#15  what a pity! Brian has to wait because the poor bank has not enough money to withdraw
#16  Audrey has deposited $325.48 @ Time 53 @ window 1
#17  Aha,Eli can withdraw enough money now so he leave the wait queue for window 2
#18  what a pity! Eli has to wait because the poor bank has not enough money to withdraw
#19  Ashley has deposited $39.66 @ Time 55 @ window 2
#20  Chase thinks queue 1 is too long,so he jumps to queue 2
#21  Chase has deposited $400.86 @ Time 62 @ window 2
#22  Aha,Amelia can withdraw enough money now so he leave the wait queue for window 2
#23  Mason has deposited $60.84 @ Time 63 @ window 1
#24  Aha,Brian can withdraw enough money now so he leave the wait queue for window 2
#25  what a pity! Brian has to wait because the poor bank has not enough money to withdraw
#26  Diego has deposited $345.98 @ Time 83 @ window 1
#27  Aha,Eli can withdraw enough money now so he leave the wait queue for window 1
********************************************************************************************

served clients :16
service Time: 122
Clients average waitting Time :6.0976
```

## 6.2   输入1000，50，30，80，4

```
Time  50/50        bankMoney $11
window 1 has 0 people:
     total waitTime
window 2 has 2 people: Bailey      Andrew
     total waitTime              20
window 3 has 2 people: Kylie       Ava
     total waitTime              23
window 4 has 1 people: Hayden
     total waitTime

waitting window :Jonathan   Lillian    Ian        Carlos      Jessica

******************************************************************************************
#1   Brandon has withdrew $60.28 @ Time 5 @ window 1
#2   what a pity! Aidan has to wait because the poor bank has not enough money to withdraw
#3   what a pity! Jonathan has to wait because the poor bank has not enough money to withdraw
#4   what a pity! Jessica has to wait because the poor bank has not enough money to withdraw
#5   Mia has deposited $21.31 @ Time 29 @ window 2
#6   Kylie has deposited $72.0 @ Time 31 @ window 3
#7   Aha,Aidan can withdraw enough money now so he leave the wait queue for window 3
#8   Owen has deposited $24.81 @ Time 32 @ window 1
#9   Aha,Jonathan can withdraw enough money now so he leave the wait queue for window 3
#10  what a pity! Jonathan has to wait because the poor bank has not enough money to withdraw
#11  Dylan has withdrew $24.1 @ Time 34 @ window 1
#12  VIP Julia gets service quickly
#13  Nicholas has deposited $17.78 @ Time 36 @ window 4
#14  what a pity! Lillian has to wait because the poor bank has not enough money to withdraw
#15  Ian thinks queue 2 is too long,so he jumps to queue 4
#16  what a pity! Ian has to wait because the poor bank has not enough money to withdraw
#17  Lauren has deposited $73.36 @ Time 41 @ window 1
#18  Aha,Jessica can withdraw enough money now so he leave the wait queue for window 4
#19  what a pity! Carlos has to wait because the poor bank has not enough money to withdraw
#20  what a pity! Jessica has to wait because the poor bank has not enough money to withdraw
#21  Madelyn has deposited $11.31 @ Time 48 @ window 1
******************************************************************************************

served clients :8
service Time: 73
Clients average waitting Time :4.3000
```

## 6.3 输入50000，60，40，300，6

```
Time  60/60      bankMoney $9422
window 1 has 2 people: Ayden      Elijah
    total waitTime               6
window 2 has 2 people: Trinity    Kimberly
    total waitTime               4
window 3 has 2 people: Brandon    Nathan
    total waitTime               2
window 4 has 2 people: Alex       Adrian
    total waitTime               1
window 5 has 2 people: Julian     Kaylee
    total waitTime               9
window 6 has 2 people: Nevaeh     Hannah
    total waitTime              24

waitting window :

*******************************************************************************************
#1  Faith has deposited $376.71 @ Time 17 @ window 6
#2  Angelina has deposited $449.3 @ Time 19 @ window 5
#3  Aaron has withdrew $487.23 @ Time 22 @ window 1
#4  Brayden has deposited $172.18 @ Time 24 @ window 3
#5  Anna has deposited $448.79 @ Time 31 @ window 4
#6  Eric has deposited $374.3 @ Time 38 @ window 2
#7  Caroline has withdrew $36.7 @ Time 40 @ window 5
#8  Alexa has deposited $308.31 @ Time 43 @ window 4
#9  Tyler has deposited $400.15 @ Time 43 @ window 5
#10 Julian thinks queue 1 is too long,so he jumps to queue 5
#11 Jackson has withdrew $463.25 @ Time 48 @ window 2
#12 Jonathan has deposited $378.64 @ Time 55 @ window 2
#13 Sara has withdrew $359.04 @ Time 56 @ window 1
#14 Madison has withdrew $126.12 @ Time 56 @ window 3
#15 Adrian has withdrew $120.66 @ Time 58 @ window 4
#16 VIP Ethan gets service quickly
*******************************************************************************************
```

# 7 附录

源文件 bankSimulate.py