

选22、抽象数据类型

1、什么是抽象数据类型

数据类型指一组性质相同的值的集合以及定义在此集合上的一些操作的总称。

抽象数据类型Abstract Data Type, ADT, 是指一个数学模型以及定义在该模型上的一组操作。

如Python自带的String、List、dict等, 除了可以存储数据外还提供了一系列相关的操作这些数据的方法。使用者在具体使用时, 不需要去了解具体的实现细节。

2、抽象数据类型的描述

ADT 抽象数据类型名称:

Data:

数据元素之间的逻辑关系定义

Operation:

操作1:

操作2:

。 。 。 。

endADT

3、通过编写链表类来实现链表的抽象数据类型（链表操作复习）

```

class linkedList:
    def __init__(self):
        self.data=[]
        self.head=-1
    def show_data(self):    #获取链表数据
        k=self.head
        res=[]
        while k!=-1:
            res.append(self.data[k][0])
            k=self.data[k][1]
        return res
    def insert(self,n,pre=None):    #在pre后插入n,若pre==None, 则插入到链表头
        if pre==None or len(self.data)==0:
            self.data.append([n,self.head])
            self.head=len(self.data)-1
        else:
            q=k=self.head
            while k!=-1:
                if self.data[k][0]==pre:
                    self.data.append([n,self.data[k][1]])
                    self.data[k][1]=len(self.data)-1
                    break
                q=k
                k=self.data[k][1]
            if k==-1:
                self.data.append([n,-1])
                self.data[q][1]=len(self.data)-1
    def delete(self,key):    #删除内容为key的节点
        q=k=self.head
        while k!=-1:
            if self.data[k][0]==key:
                if k==self.head:
                    self.head=self.data[k][1]
                else:
                    self.data[q][1]=self.data[k][1]
                    self.data[k]=None
                break
            q=k
            k=self.data[k][1]

```

```

la=linkedList()
la.insert(3,1)
la.insert(5,3)
la.insert(4,5)
la.insert(2,None)
la.insert(7,1)
print(la.show_data())
la.delete(4)
la.delete(2)
la.delete(6)    #如果没有该节点, 就不会有任何操作
print(la.show_data())

```

```

[2, 3, 5, 4, 7]
[3, 5, 7]

```