改变参数值的函数

礼 欣

北京理工大学

6.5 改变参数值的函数

- 函数通过参数与调用程序传递信息。
- 例子1：银行账户计算利率
  - 账户余额计算利息的函数：

```
# addinterest1.py
def addInterest(balance, rate):
    newBalance = balance * (1+rate)
    balance = newBalance
def main():
    amount = 1000
    rate = 0.05
    addInterest(amount, rate)
    print(amount)

main()
```

6.5 改变参数值的函数

- 期待效果：为余额1000的账户上增加5%，那么账户余额变为1050
- 然而实际输出为。。。

```
>>> main()
1000
```

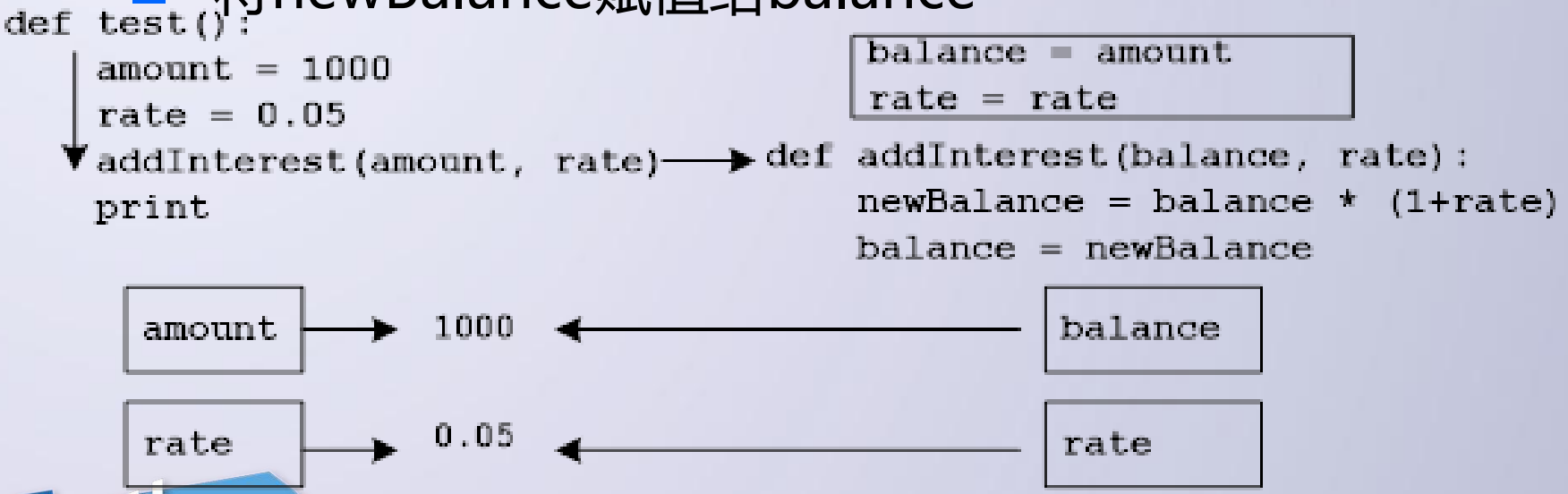
python 语言程序设计

6.5 改变参数值的函数

- 注意：虽然形参和实参名字相同，但他们是不同的变量
- 参数赋值使得addInterest()中变量amount和rate引用了实参的值

6.5 改变参数值的函数

- 分析addInterest()的调用过程：
  - 将newBalance赋值给balance



python 语言程序设计

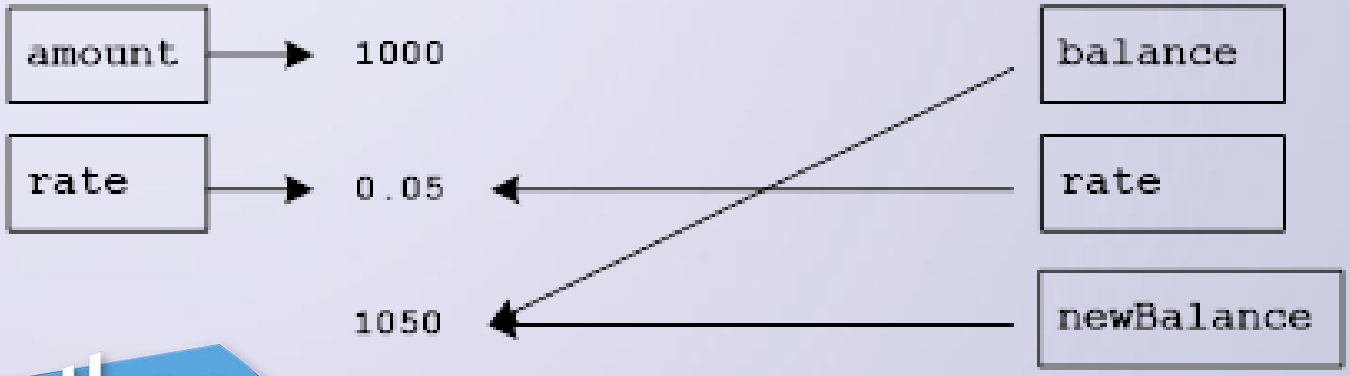
6.5 改变参数值的函数

- balance的赋值过程：

```
def test():
    amount = 1000
    rate = 0.05
    addInterest(amount, rate)
    print
```

```
balance = amount
rate = rate
```

```
def addInterest(balance, rate):
    newBalance = balance * (1+rate)
    balance = newBalance
```



python 语言程序设计

6.5 改变参数值的函数

- 分析可知：函数的形参只接收了实参的值，给形参赋值并不影响实参。
- Python可以通过值来传递参数。
- 方法：改变函数addInterest() 返回一个newBalance，用这个值更新test()中的amount

6.5 改变参数值的函数

- 代码：

```
# addinterest2.py

def addInterest(balance, rate):
    newBalance = balance * (1+rate)
    return newBalance
def test():
    amount = 1000
    rate = 0.05
    amount = addInterest(amount, rate)
    print(amount)
test()
```

python 语言程序设计

- 运行结果：1050.0

6.5 改变参数值的函数

- 例子2：处理多个银行帐户的程序
- 实现：
  - 用列表存储帐户余额信息，列表中的第一个帐户余额：

```
balances[0] = balances[0] * (1 + rate)
```

  - 下一个位置的值：

```
balances[1] = balances[1] * (1 + rate)
```

6.5 改变参数值的函数

- 对列表位置0,1,… ,length - 1 的值使用循环进行余额计算
- 代码：

```
# addInterest3.py

def addInterest(balances, rate):
    for i in range(len(balances)):
        balances[i] = balances[i] * (1+rate)
def test():
    amounts = [1000, 105, 3500, 739]
    rate = 0.05
    addInterest(amounts, rate)
    print(amounts)
test()
```

6.5 改变参数值的函数

- 运行结果：

```
>>>
[1050.0, 110.25, 3675.0, 775.95]
```

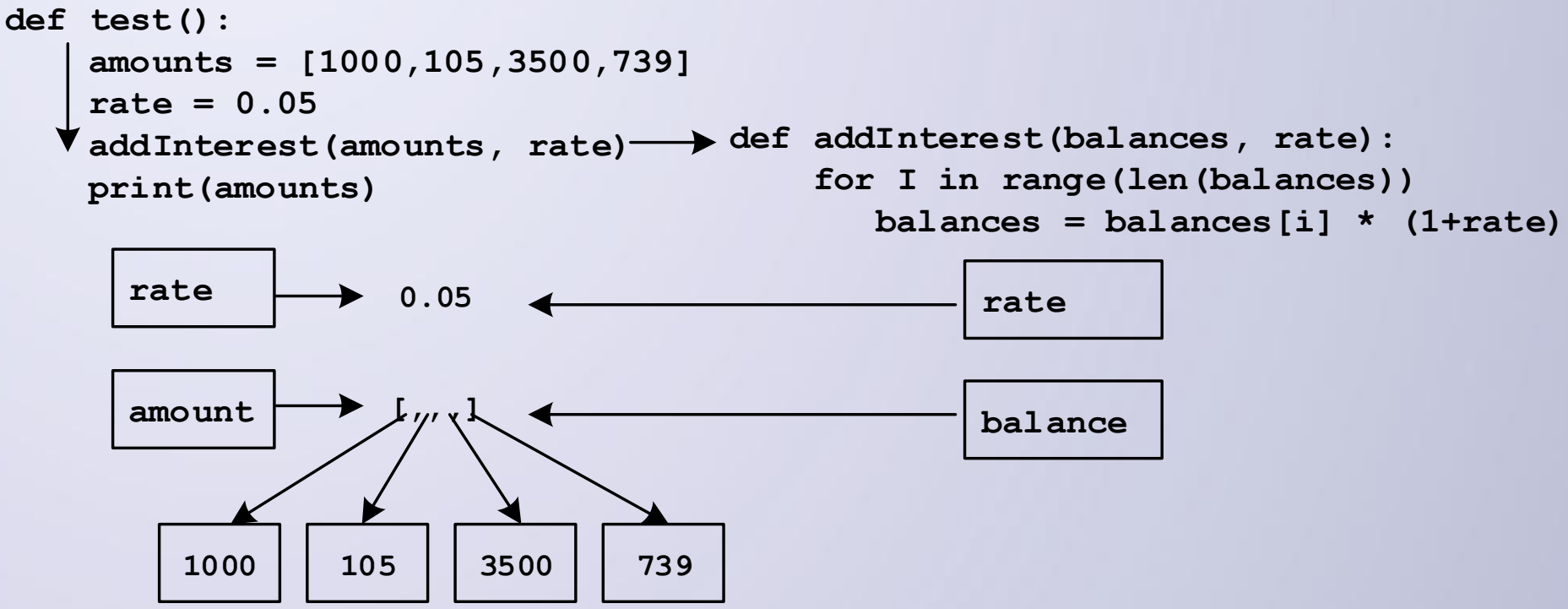
6.5 改变参数值的函数

- 函数不能修改变量本身（即amounts）
- Test()中amounts是一个包含四个整数类型值的列表对象，以实参的形式传递给函数addInterest()的形参balances。

python 语言程序设计

6.5 改变参数值的函数

- test()向函数addIntertest()传递列表参数图

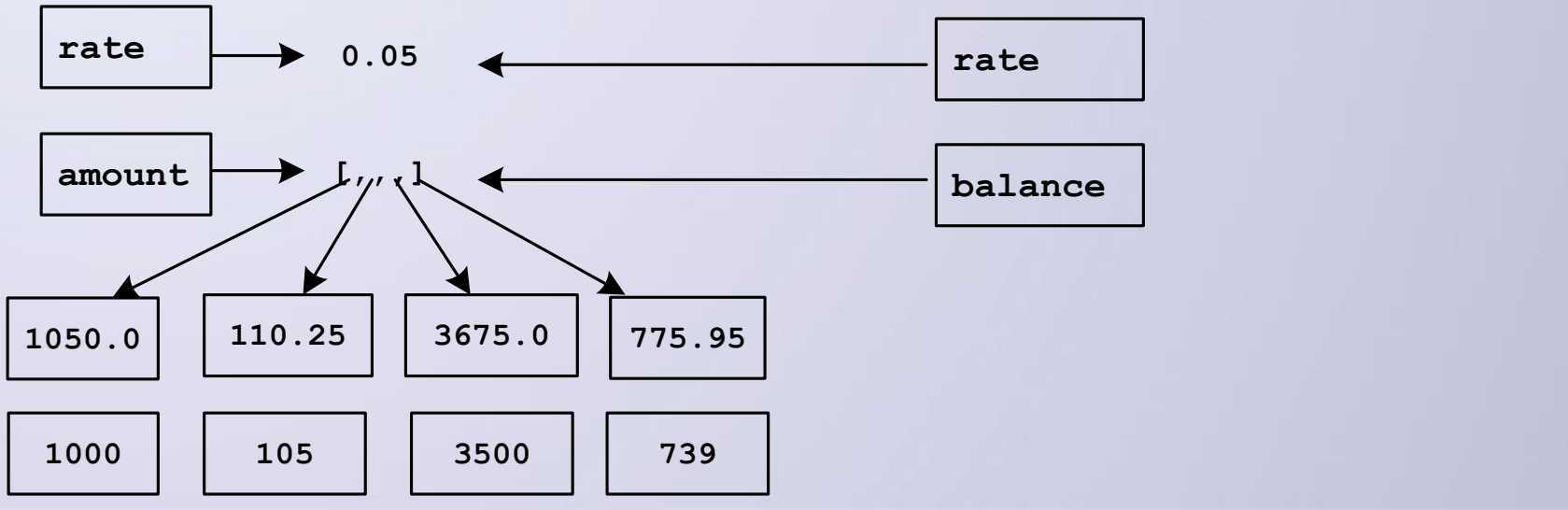


6.5 改变参数值的函数

- addIntertest()中列表被修改：从0到length-1范围执行循环，并更新balances的值

```
def test():
    amounts = [1000,105,3500,739]
    rate = 0.05
    addInterest(amounts, rate)
    print(amounts)
```

```
def addInterest(balances, rate):
    for I in range(len(balances))
        balances = balances[i] * (1+rate)
```



6.5 改变参数值的函数

- 变量amounts**没有改变，它仍然指向的是调用函数addInterest()之前同一个列表**
- 函数addInterest()结束时，存储在amounts中的列表存储了新的balances值
  - 旧值在Python垃圾数据回收的时候被清除掉

python 语言程序设计

6.5 改变参数值的函数

- 总结：Python的参数是通过值来传递的
- 如果变量是可变对象（如列表或者图形对象），返回到调用程序后，该对象会呈现被修改后的状态。