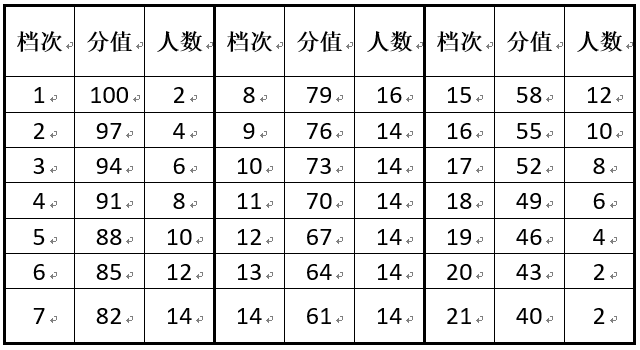
**从VB到Python之选考成绩赋分**

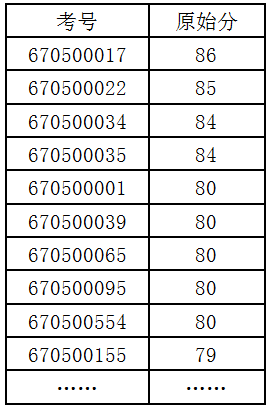
余姚二中 梁见斌

VB代码：

17．【**加试题**】(2018年12月浙江省重点中学期末热身联考)成绩赋分。浙江省的选考成绩采用等级赋分制，共分为21档。赋分方法是先将该科目的成绩从高到低排序，再依次根据不同档次的人数进行赋分。当成绩相同但人数超出该等级人数时，则减少下一档的人数。当原始成绩为0分，则赋分成绩也为0分。某学校某门科目的选考人数为200人，部分成绩如第17题图a所示，其相应等级的人数如第17题图b所示。请回答下列问题：



第17题图b



第17题图a

（1）根据题意，观察赋分表，考号670500554的学生的赋分成绩是 ▲ 。

（2）实现上述功能的程序如下，请在划线处填上合适的代码。

Dim a(1 To 21) As Integer '该档次赋分分值

Dim b(1 To 21) As Integer '该档次人数

Dim kh(1 To 200) As String '学生考号

Dim cj1(1 To 200) As Integer '学生的原始成绩

Dim cj2(1 To 200) As Integer '学生赋分后的成绩

Private Sub Form\_Load()

'从数据库中读取数据，将21个等级的分值和人数存储在数组a和数组b中，将学生的考号和原始成绩存储在数组kh和数组cj1中。最终根据原始成绩从高到低排序后将kh和cj1输出在列表框List1中，代码略。

End Sub

Private Sub Command1\_Click()

Dim k As Integer

Dim flag As Boolean '是否还可以对该档次进行赋分

k = 1: t = 1: cj2(1) = a(1): flag = True

For i = 2 To 200 '根据学生的原始成绩按档次进行赋分

If cj1(i) = 0 Then

cj2(i) = 0

ElseIf ① Then

t = t + 1: cj2(i) = a(k)

Else

flag = False

End If

If Not flag Then '当前档次的分值已经赋分完毕

Do While t >= b(k)

②: k = k + 1

Loop

t = t + 1: cj2(i) = a(k): ③

End If

Next i

List2.AddItem "学生考号" + " " + "学生成绩"

For i = 1 To 200

List2.AddItem kh(i) + " " + Str(cj2(i))

Next i

End Sub

答案：（1）97 （1分）

（2）①cj1(i) = cj1(i - 1) Or t < b(k) （2分）

②t = t - b(k) 或 b(k + 1) = b(k + 1) + b(k) （2分）

③flag = True （1分）

算法分析：最直接的思路是先根据已经按降序排列的原始成绩，计算出对应的排名（成绩相同的排名也相同），存储到数组mc；然后根据各档学生人数计算到第i档为止的人数，存储到数组b2。这样mc和b2就一一对应起来了，可以直接根据它们的关系对学生进行赋分。

本题的VB代码没有采用上述分步走的思路，而是把计算排名，更新各档次人数和赋分放到一起来做了。使用Python语言来表达上述算法思想代码会更简洁些。

对应Python代码：

#算法1：分步走，先求排名和到第i档为止的人数，再计算赋分

def grade\_1(a, b, c):

mc = [0] \* len(c) #根据学生原始成绩，计算学生排名

mc[1] = 1

for i in range(2,len(c)):

if c[i] != c[i-1]:

mc[i] = i

else:

mc[i] = mc[i-1]

b2 = list(b) #根据各档学生人数计算到第i档为止的人数

for i in range(2,len(b)):

b2[i] += b2[i-1]

c2 = ["赋分"]

k, i, n = 1, 1, len(c1)

while i < n: #计算学生赋分

if mc[i] <= b2[k]:

c2.append(a[k])

i += 1

else:

k += 1

i = n - 1

while c[i] == 0: #为0分的学生赋分

c2[i] = 0

i -= 1

return c2

#算法2：把计算排名，更新各档次人数和赋分放到一起处理

def grade\_2(a, b, c):

b2 = list(b) #需要将第i档学生人数更新为到第i档为止的人数

c2 = ["赋分"]

c2.append(a[1])

k, i, n = 1, 2, len(c)

while i < n and c[i] != 0: #为非0分的学生赋分

if i <= b2[k] or c[i] == c[i-1]:

c2.append(a[k])

i += 1

else:

b2[k+1] += b2[k]

k += 1

while i < n: #为0分的学生赋分

c2.append(0)

i += 1

return c2

主函数的功能上从excel表格读取数据，调用计算赋分的函数grade()后，再把生成的赋分写入excel表格。

from openpyxl import load\_workbook

#读取赋分表和学生原始成绩

wb\_r = load\_workbook("选考成绩赋分.xlsx")

print(wb\_r.sheetnames)

ws = wb\_r['赋分表']

a = tuple(map(lambda x: x.value, ws['B']))#各档次赋分分值

b = tuple(map(lambda x: x.value, ws['C']))#各档次人数

ws = wb\_r['学生信息']

c1 = tuple(map(lambda x: x.value, ws['D']))#原始成绩，已按降序排列

c2 = grade\_1(a, b, c1) #赋分成绩

#将赋分成绩写入“学生信息”表的“赋分”列

for i in range(len(c2)):

ws[f'E{i+1}'] = c2[i]

wb\_r.save("选考成绩赋分.xlsx")

上述两种算法，grade\_1()是直接模拟手工计算赋分的过程，分成3步走，先根据已排序的原始成绩计算排名，再根据根据各档学生人数计算到第i档为止的人数，最后在生成赋分成绩，思路简洁明了，条理清晰，满足Python语言“简洁而不是简短”的训诫；grade\_2()把3步并成1步，代码更简洁，但是理解上稍有难度，需要认真分析才能体会它的精妙之处。