贪婪搜索——贪婪算法

一、 实验目的：

学习用python来编写贪婪算法

二、 实验内容：

现在有很多箱重量和价值都不相同的货物，每箱货物都可以分成任意散装组合运送。货车最多只能承受一定重量的货物。要求能运送的货物的最大总价值

可以对货物进行规定次序的排序，依次放入高价值货物后判断是否超出最大承重，超出则要依次减小重量知道找到最优的方案。

三、 实验代码

from \_\_future\_\_ import division

input x=input(**'箱数:'**)  
input\_y = input(**'最大承受重量:'**)  
list\_a = []

list\_b = []

for i in range(1,int(input\_x)+1):

input\_Tvalue = input('第'+str(i)+'箱的总价值:')

input\_Theav = input('第'+str(i)+'箱的重量:')

avg = round(int(input\_Tvalue)/int(input\_Theav),1)#每一箱单位重量的价值

list\_a.append(avg)#添加到列表，用于比较

list\_b.append([int(input\_Theav),avg,0]) list\_a.sort(reverse=True) # 降序排序

sum =[0,0num =0

full = 0

for i in range(len(list\_a)):

for k in range(len(list\_b)):

if full == 0

if (list\_a[i] == list\_b[k][1]) and (list\_b[k][2]==0):

sum[1] = sum[0]

sum[0] = sum[0] + list\_b[k][0]

v = list\_b[k][0]

if sum[0] > int(input\_h)

full = 1

t= list\_b[k][0]

while True:

z = sum[1] + t

if z <= int(input\_h):

break

t = t-1

v=t

sum[0]=sum[1]

sum[0] = sum[0] + t

num = list\_a[i]\*v + num

list\_b[k][2] = 1

print ('能带走的货物的最大价值为:',num)

五、 实验结果：



