Problem 4

(a)

先依照大小排序，由大排到小(最大的還是編號設為1，但是和題目的編號不同)。假設第j個是此algorithm拒絕的item。

假設0/1 knapsack的optimal solution為OPT, 用此algorithm的解為A，fractional knapsack的optimal solution為

要證明:

, 其中

由於fractional knapsack不可能比0/1 knapsack的optiaml solution還差，可知

...…(2)

Why?

如果

如果

如果

由(1), (2)可知

(b)

DP (i, V ) is the minimum weight with total value at least V within the first i items.

Recursive formula: 🡪in small integer value this will give an exact solution!

沒選第i個item和有選第i個item

1

2

3 for i from 1 to n //total n items, O(n)

4 for each possible v from 1 to //

5

6 for k from //

7 if DP(n, k)<=W return k

填表格有兩個for loop花O(n)，最後找答案一個for loop花，一共花O()

(c)

1 找一個K，並且把所有的value都reduce成floor(v/K)，形成v’ array //O(n)

2 利用新的v’，代入(b)的dynammic programming algorithm算出ans //

3 return ans\*K

因為所有的value都被reduced過，所以算出來的optimal value也會接近，所以第2步算DP table的時間複雜度會變為O()。第1步算reduced value花O(n)，所以一共花O(n)+O()= O()。

(d)

假設S\*為原本0/1 knapsack problem的最佳解集合，S’\*為0/1 knapsack problem with reduced values的最佳解集合，c(S\*)為S\*用原value求出的總值，為原本問題中的optimal value，c’(S\*)為S\*中用reduced values求出的總值。

如此c(S\*)=V\*, c’(S’\*)=

由(c)所傳回的解為K\* c’(S’\*)

0/1 knapsack problem with reduced values的optimal value，至少會比其他解的value大，包含

由(1)，可推得

其中

Why?

又已知c(S\*)=V\*, c’(S’\*)=，代入(3)，得

……(4)

又

由(4), (5)可得

證明完畢。