Problem 2

a) FRACTIONAL-KNAPSACK (n, w1, w2, …, wn , v1, v2, …, vn)

calculate the ratio value/weight for each item through vi/wi for all i, i=1,2,3,…n

sort the items in the descending of the calculated ratios and renumber such that

v1/w1 ≥ v2/w2 ≥ ….. ≥ vn/wn

```
knapsack = []
capacity ← C
FOR i = 1 to n
    IF wi <= C
        Add ai to knapsack
        C = C – wi
    ELSE
        Add (C/wi)*ai to knapsack
        C = 0
    IF C == 0
        Return knapsack
```

b) The greedy algorithm will not find an optimal solution for the binary knapsack problem.

Let’s look at a counter example where these are the items and capacity of knapsack is 11

i	V_i	W_i	V_i/W_i
1	1	1	1
2	6	2	3
3	18	5	3.6
4	22	6	3.7
5	28	7	4

Now following the greedy algorithm approach,

item 5 will be selected, capacity remaining = 11-7 = 4

Then item 2 will be selected, capacity remaining = 4-2 = 2

Then item 1 will be selected, capacity remaining 2-1 =1

But since no further item is available, the algorithm will terminate.

The solution will be: 5, 2, 1 with total value = 28+6+1 = 35

However, the optimal solution to this problem is 6, 5 with total value = 22+18 = 40

Hence, the greedy algorithm fails to provide optimal solution in the case of binary knapsack problem.