mini hw-4

Given N items, where the i-th item is with value a_i, volume v_i and weight w_i $(a_i, v_i, w_i \in \mathbb{N} \cup \{0\})$. Please give an $O(NVW)$ algorithm to find a subset S of these items such that:

- the total volume of the items in S does not exceed V
- the total weight of the items in S does not exceed W
- the total value of the items in S is maximized

Briefly explain the workflow of your algorithm. You do **not** need to prove its correctness in this problem.

Optimal substructure: suppose OPT is an optimal solution to this knapsack problem $KP(i, v, w) KP(i)$, for i-th item, there is 2 cases:

1. Case 1: not to choose the i-th item in OPT
    a. OPT is an optimal solution of $KP(i-1, v, w)$
2. Case 2: choose the i-th item in OPT
    a. OPT\\i-th item is an optimal solution of $KP(i-1, v-v_i, w-w_i)$

Recursively define the value:

$$M_{i,v,w} = \begin{cases} 0, \text{if } i = 0 \\ M_{i-1,v,w}, \text{if } v_i > v || w_i > w \\ \max(M_{i-1,v,w}, a_i + M_{i-1,v-v_i,w-w_i}), \text{otherwise} \end{cases}$$

We can use Bottom-up or Top-down method to compute values.

Then, we can backtrack the solution S by following method.

```
Find Solution(M,i,V,W)
  S = {}. //set of solution
  v=V
  w=W
  for i = 1 to n
    if M[i,v,w] == M[i - 1, v ,w] //find the path
      w = w - w[i]
      v = v - v[i]
      S = S U {i}
return S
```