```
[2] from random import randint

N = [i for i in range(50, 1001, 50)]

A = list()
for n in N:
   temp = list()
   while len(temp) < n:
        a = randint(0, 500)
        temp.append(a)
        A.append(temp)</pre>
```

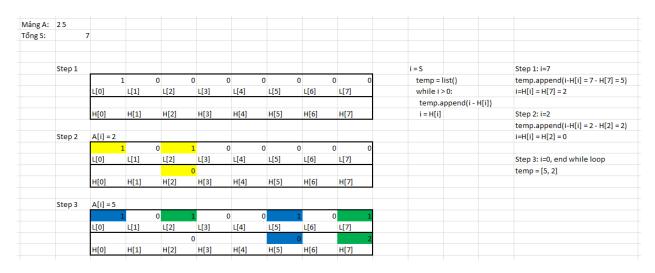
Tạo mảng A chứa các mảng có lần lượt 50, 100, ..., 1000 phần tử ngẫu nhiên trong khoảng (0, 500) có thể có phần tử giống nhau.

```
[10] def knapsack(A, S):
       ## Mảng L cho biết có tính được tổng S không? Nếu L[S] = 1 thì được, L[0] !=1 thì không.
       L = [0] * (S + 1)
       L[0] = 1
       ## Mảng H dùng để truy vết. Để tính được tổng S thì ta phải có tổng H[S].
       H = [0] * (S + 1)
       ## Fill mång L và H
       for i in range(len(A)):
         for j in range(S, A[i] - 1, -1):
           if (L[j] == 0) and (L[j - A[i]] == 1):
             L[j] = 1
             H[j] = j - A[i]
       if L[S] != 1: return -1
         i = S
         temp = list()
         while i > 0:
           temp.append(i - H[i])
           i = H[i]
       return temp
```

Ý tưởng về thuật toán, ta sẽ dùng mảng L có độ dài S+1 để lưu hai số 0 hoặc 1. Giá trị của index sẽ đại diện cho tổng mà ta xét, còn L[index] sẽ cho ta biết là tổng đang xét có tính được hay không. Ví dụ L[10] = 1, tức là có thể tính được tổng 10 với mảng đầu vào. Index 0 sẽ được gán là 1.

Ta sẽ dùng 2 vòng lặp for để duyệt mảng số đầu vào có N phần tử, duyệt ngược mảng L (từ tổng lớn đến tổng nhỏ) có S+1 phần tử, tuy nhiên ta sẽ không duyệt L[0], vậy độ phức tạp thời gian sẽ là N*S. Nếu L[j] = 0 tức tổng bằng j chưa tính được, và L[j - A[i]] = 1 tức tổng bằng j - A[i] đã tính được, khi đó ta gán L[j] = 1 tức đã tính được tổng bằng j. Khi đó ta cũng gán H[j] = j - A[i].

Ví dụ, với A[i] = 7, nếu có tổng 3 tức L[3] = 1, thì ta sẽ có tổng 10, vậy ta gán L[10] = 1. Khi đó, ta cũng gán H[10] = 10 - 7 = 3. Mảng H sẽ giúp ta truy ra mảng con đầu ra mà đề bài yêu cầu. Mảng H theo lí thuyết thì ban đầu sẽ là mảng rỗng có độ dài S+1, tuy nhiên khi code thì ta không tạo mảng như vậy được, nên ở đây ta cho nó là mảng 0 (hoặc số âm tùy chọn).



Minh họa một ví dụ đơn giản trên excel.

```
S = 200
for a in A:
    print(f"A = {set(a)}\nSubset = {knapsack(list(set(a)), S)}\n")
```

Chạy thuật toán với mỗi mảng trong A và in ra kết quả. Một điều cần chú ý, do mảng số đầu vào có thể có phần tử giống nhau nên thuật toán có thể trả về mảng con có phần tử giống nhau. Tuy nhiên, đề bài yêu cầu mảng con không được có phần tử giống nhau. Do đó, ta phải dùng set() để loại bỏ các phần tử giống nhau trong mảng, rồi chồng thêm list() để chuyển về dạng list rồi mới đưa vào chương trình để chạy.