```
def knapsack(value, weight, capacity):
    """Return the maximum value of items that doesn't exceed capacity.

    value[i] is the value of item i and weight[i] is the weight of item i
    for 1 <= i <= n where n is the number of items.

    capacity is the maximum weight.
    """
    n = len(value) - 1

    # m[i][w] will store the maximum value that can be attained with a maximum
    # capacity of w and using only the first i items
    m = [[-1]*(capacity + 1) for _ in range(n + 1)]

    for w in range(capacity + 1):
        m[0][w] = 0

    for i in range(1, n + 1):
        for w in range(capacity + 1):
            if weight[i] > w:
                m[i][w] = m[i - 1][w]
            else:
                m[i][w] = max(m[i - 1][w - weight[i]] + value[i],
                              m[i - 1][w])

    return m[n][capacity]


n = int(input('Enter number of items: '))
value = input('Enter the values of the {} item(s) in order: '
              .format(n)).split()
value = [int(v) for v in value]
value.insert(0, None) # so that the value of the ith item is at value[i]
weight = input('Enter the positive weights of the {} item(s) in order: '
               .format(n)).split()
weight = [int(w) for w in weight]
weight.insert(0, None) # so that the weight of the ith item is at weight[i]
capacity = int(input('Enter maximum weight: '))

ans = knapsack(value, weight, capacity)
print('The maximum value of items that can be carried:', ans)
```