SUBSET – SUM PROBLEM

PROBLEM & GOAL

- We are given **n** items {1, . . . , n}, and each has a given nonnegative weight $\mathbf{w_i}$ (for i = 1, . . . , n).
- We are also given a bound W.
- We would like to select a subset S of the items so that $\sum_{i \in S} \mathbf{w_i} \leq \mathbf{W}$ and, subject to this restriction, $\sum_{i \in S} \mathbf{Wi}$ is as large as possible.
- We will call this the Subset Sum Problem.

DESIGNING THE ALGORITHM

- Let us consider and optimal solution.
- **OPT(n, W) = max profit subset of items 1, …, n with weight limit W.**
- There can be 2 cases if we consider an nth item as follows
  - **Case 1: OPT** does not select item n i.e. n $\notin$ OPT
    - **OPT** selects **best** of **{ 1, 2, …, n-1 }** using weight limit **W**.
  - **Case 2: OPT** selects item n. (i.e. n $\in$ OPT)
    - New weight limit = **$W - w_n$**
    - **OPT** selects best of **{ 1, 2, …, n–1 }** using **this new weight limit**.

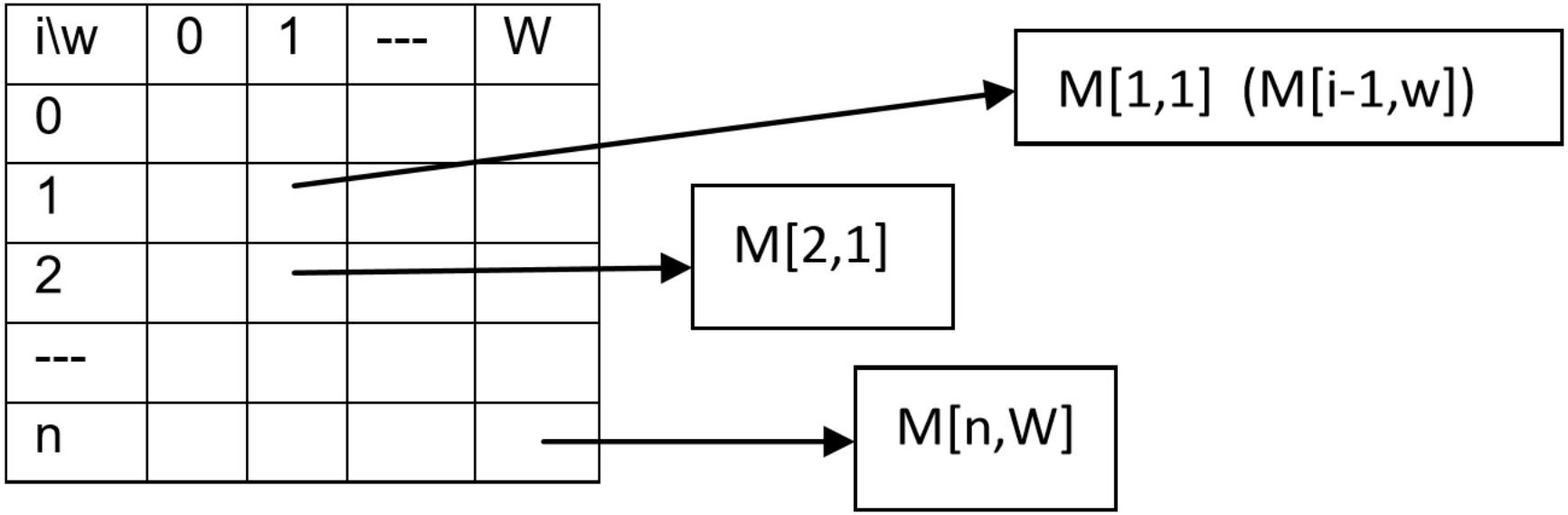
DESIGNING THE ALGORITHM

- Recurrence Relation

$$OPT(n,W) = \begin{cases} 0 & \text{if } n = 0 \\ OPT(n-1,W) & \text{if } w_n > W \\ \max\{OPT(n-1,W), w_n + OPT(n-1,W-w_n)\} & \text{otherwise} \end{cases}$$

DESIGNING THE ALGORITHM

- We need two dimensional table of (n,W) to fill in the values in order to get the maximum subset of weights of the items.



SUBSET – SUM ALGORITHM

//Purpose: To find the maximum weight from the given n items and their weights wi

//Input: A set of items 1,2,…….n, with, w1, … ,wN, capacity W

//Output: Max weight M[n, W]

```
for w = 0 to W
   M[0, w] = 0
for i = 0 to n
   M[i, 0] = 0
for i = 1 to n                    // n items
   for w = 1 to W                 // weights from 1 to max cap W
      if (wi > w)
         M[i, w] = M[i-1, w]
      else
         M[i, w] = max {M[i-1, w], wi + M[i-1, w-wi] }
   endfor
endfor
return M[n, W]
```

SUBSET – SUM EXAMPLE

ITEMS	WEIGHTS	
1	2	W = 5
2	1	
3	3	

	0	1	2	3	4	5
0	0	0	0	0	0	0
1	0					
2	0					
3	0					

SUBSET – SUM EXAMPLE

i	wi	w	wi > w	M [i, w]
1	2	1	2 > 1 **TRUE**	**= M[i-1, w]** **= M[0, 1]** **= 0**
		2	2 > 2 **FALSE**	**= Max{ M[i-1, w], wi + M[i-1, w-wi] }** **= Max (M [0 , 2] , 2 + M [0 , 0])** **= Max(0,2)** **= 2**

	0	1	2	3	4	5
0	**0**	**0**	**0**	**0**	**0**	**0**
1	**0**	**0**	**2**			
2	**0**					
3	**0**					

SUBSET – SUM EXAMPLE

i	wi	w	wi > w	M [i, w]
1	2	3	2 > 3 **FALSE**	**= Max{ M[i-1, w], wi + M[i-1, w-wi] }** **= Max (M [0 , 3] , 2 + M [0 , 1])** **= Max (0 , 2)** **= 2**
		4	2 > 4 **FALSE**	**= Max{ M[i-1, w], wi + M[i-1, w-wi] }** **= Max (M [0 , 4] , 2 + M [0 , 2])** **= Max (0 , 2)** **= 2**

	0	1	2	3	4	5
0	**0**	**0**	**0**	**0**	**0**	**0**
1	**0**	**0**	**2**	**2**	**2**	
2	**0**					
3	**0**					

SUBSET – SUM EXAMPLE

i	wi	w	wi > w	M [i, w]
1	2	5	2 > 5 **FALSE**	**= Max{ M[i-1, w], wi + M[i-1, w-wi] }** **= Max (M [0 , 5] , 2 + M [0 , 3])** **= Max (0 , 2)** **= 2**

	0	1	2	3	4	5
0	**0**	**0**	**0**	**0**	**0**	**0**
1	**0**	**0**	**2**	**2**	**2**	**2**
2	**0**					
3	**0**					

SUBSET – SUM EXAMPLE

i	wi	w	wi > w	M [i, w]
2	1	1	1 > 1 **FALSE**	**= Max{ M[i-1, w], wi + M[i-1, w-wi] }** **= Max (M [1 , 1] , 1 + M [1 , 0])** **= Max (0 , 1)** **= 1**
		2	1 > 2 **FALSE**	**= Max{ M[i-1, w], wi + M[i-1, w-wi] }** **= Max (M [1 , 2] , 1 + M [1 , 1])** **= Max (2 , 1)** **= 2**

	0	1	2	3	4	5
0	**0**	**0**	**0**	**0**	**0**	**0**
1	**0**	**0**	**2**	**2**	**2**	**2**
2	**0**	**1**	**2**			
3	**0**					

SUBSET – SUM EXAMPLE

i	wi	w	wi > w	M [i, w]
2	1	3	1 > 3 **FALSE**	**= Max{ M[i-1, w], wi + M[i-1, w-wi] }** **= Max (M [1 , 3] , 1 + M [1 , 2])** **= Max (2 , 3)** **= 3**
		4	1 > 4 **FALSE**	**= Max{ M[i-1, w], wi + M[i-1, w-wi] }** **= Max (M [1 , 4] , 1 + M [1 , 3])** **= Max (2 , 3)** **= 2**

	0	1	2	3	4	5
0	0	0	0	0	0	0
1	0	0	2	2	2	2
2	0	1	2	3	3	
3	0					

SUBSET – SUM EXAMPLE

i	wi	w	wi > w	M [i, w]
2	1	5	1 > 5 **FALSE**	**= Max{ M[i-1, w], wi + M[i-1, w-wi] }** **= Max (M [1 , 5] , 1 + M [1 , 4])** **= Max (2 , 3)** **= 3**

	0	1	2	3	4	5
0	**0**	**0**	**0**	**0**	**0**	**0**
1	**0**	**0**	**2**	**2**	**2**	**2**
2	**0**	**1**	**2**	**3**	**3**	**3**
3	**0**					

SUBSET – SUM EXAMPLE

i	wi	w	wi > w	M [i, w]
3	3	1	3 > 1 **TRUE**	**= M[i-1, w]** **= M [i – 1 , w]** **= M [3 – 1 , 1]** **= 1**
		2	3 > 2 **TRUE**	**= M[i-1, w]** **= M [i – 1 , w]** **= M [3 – 1 , 2]** **= 2**

	0	1	2	3	4	5
0	**0**	**0**	**0**	**0**	**0**	**0**
1	**0**	**0**	**2**	**2**	**2**	**2**
2	**0**	**1**	**2**	**3**	**3**	**3**
3	**0**	**1**	**2**			

SUBSET – SUM EXAMPLE

i	wi	w	wi > w	M [i, w]
3	3	3	3 > 3 **FALSE**	**= Max{ M[i-1, w], wi + M[i-1, w-wi] }** **= Max (M [2 , 3] , 3 + M [2 , 0])** **= Max (3 , 3)** **= 3**
		4	3 > 4 **FALSE**	**= Max{ M[i-1, w], wi + M[i-1, w-wi] }** **= Max (M [2 , 4] , 3 + M [2 , 1])** **= Max (3 , 4)** **= 4**

	0	1	2	3	4	5
0	0	0	0	0	0	0
1	0	0	2	2	2	2
2	0	1	2	3	3	3
3	0	1	2	3	4	

SUBSET – SUM EXAMPLE

i	wi	w	wi > w	M [i, w]
3	3	3	3 > 5 **FALSE**	**= Max{ M[i-1, w], wi + M[i-1, w-wi] }** **= Max (M [2 , 5] , 3 + M [2 , 2])** **= Max (3 , 5)** **= 5**

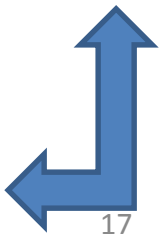
	0	1	2	3	4	5
0	**0**	**0**	**0**	**0**	**0**	**0**
1	**0**	**0**	**2**	**2**	**2**	**2**
2	**0**	**1**	**2**	**3**	**3**	**3**
3	**0**	**1**	**2**	**3**	**4**	**5**

SUBSET – SUM EXAMPLE

ITEMS	WEIGHTS	
1	2	W = 5
2	1	
3	3	

	0	1	2	3	4	5
0	0	0	0	0	0	0
1	0	0	2	2	2	2
2	0	1	2	3	3	3
3	0	1	2	3	4	5

SOLUTION



THANK YOU