Ayush Kumar Gupta 19010201 DATE: / / PAGE NO. B+

Subset sum problem is as follows:-

⇒ Given some natural numbers $w_1, w_2, \dots w_n$ and a target number W, is there a subset of $\{w_1, w_2, \dots w_n\}$ that adds up to W?

example:-

Let the set have $\{3, 34, 4, 12, 5, 2\}$

$W = 9$

$\Rightarrow \{4, 5\}$ is a subset which adds up to 9

We can use dynamic programming to solve this problem.

We will create a 2D array Mat of size $(n+1) * (W+1)$ of bool type. The state Mat[i][j] will be true if there exists a subset of elements from $w[0 \dots i]$ with sum value = 'j'.

The pseudocode for recursion is as follows:-

```
if (w[i] > j)
    Mat[i][j] = Mat[i-1][j]
```

else

```
Mat[i][j] = Mat[i-1][j]  or  Mat[i-1][j-w[i]]
```

This means that if current element has value greater than 'current sum value' we will copy answer from previous cases

And if the current sum value is greater than the 'ith' element we will see if any of previous states have already experienced the sum = 'j' or any previous states experienced a value 'j - wt[i]' which will solve our purpose.

Here is a simulation:

wt set = {3, 4, 5, 2} W = 6

Mat =>

	0	1	2	3	4	5	6
0	T	F	F	F	F	F	F
3	T	F	F	T	F	F	F
4	T	F	F	T	T	F	F
5	T	F	F	T	T	T	F
2	T	F	T	T	T	T	T