The Subset-Sum problem

- Input: An array A[1 . . . n] of positive integers, and a target integer T > 0.
- Output: True iff there is a subset of the A values that sums to T .
- Example
  - Input: A = [1, 3, 4, 6, 10] and T = 16
  - Output: True
- If we look at all subsets for a sum of T then the running time is exponential.

The Subset-Sum problem

Outline:

Let $S[i, j]$ be defined as true iff there is a subset of elements $A[1 \ldots i]$ that sums to j .

Then $S[n, T]$ is the solution to our problem.

In general:

$$S[i, j] = S[i - 1, j - A[i]] \vee S[i - 1, j]$$

With initial conditions

$$S[i, 0] = \text{True, and } S[0, j] = \text{False, for } j > 0.$$

$A = [2, 3, 7, 9, 10] \qquad T = 11$

$$
\begin{array}{cc|ccccccccccc}
 & & & & & & T & & & & & & \\
 & & 1 & 2 & 3 & 4 & 5 & 6 & 7 & 8 & 9 & 10 & 11 \\
\hline
 & 1 - 2 & F & T & F & F & F & F & F & F & F & F & F \\
 & 2 - 3 & F & T & T & F & T & F & F & F & F & F & F \\
n & 3 - 7 & F & T & T & F & T & F & T & F & T & T & F \\
 & 4 - 9 & F & T & T & F & T & F & T & F & T & T & T \\
 & 5 - 10 & F & T & T & F & T & F & T & F & T & T & T
\end{array}
$$

Translating to pseudo-code

```
for (i = 0; i <= n; i ++)
    S [i, 0]= TRUE;
for (j = 1; j <= T; j ++)
    S [0,j] = FALSE;
for (i = 1; i <= n; i ++)
    for (j = 1; j <= T; j ++)
        if j < A[i] then
            S [i,j] = S [i -1, j];
        else
            S [i,j] = S [i -1, j] || S [i-1, j-A[i]]);
```

Time complexity is $\Theta(nT)$.