**KNA PSACK (repetition)**

**INPUT:** A set of items with (weights, values) 
\((w_1, v_1), (w_2, v_2), \ldots, (w_n, v_n)\)

**TOTAL WEIGHT = W**

**GOAL:** Find a subset of items with Total weight < W and maximum value

**W = 30**

\(\{A, A\}\)

\[15 + 15 \leq 30\]

\[43 + 43 = 86\]

\(\{A, C, D\}\)

\[15 + 7 + 8 \leq 30\]

\[43 + 19 + 23 = 84\]

<table>
<thead>
<tr>
<th>ITEM</th>
<th>WEIGHT</th>
<th>VALUE</th>
</tr>
</thead>
<tbody>
<tr>
<td>A</td>
<td>15</td>
<td>43</td>
</tr>
<tr>
<td>B</td>
<td>6</td>
<td>18</td>
</tr>
<tr>
<td>C</td>
<td>7</td>
<td>19</td>
</tr>
<tr>
<td>D</td>
<td>8</td>
<td>23</td>
</tr>
<tr>
<td>E</td>
<td>5</td>
<td>14</td>
</tr>
<tr>
<td>F</td>
<td>4</td>
<td>9</td>
</tr>
</tbody>
</table>
KNAPSACK (no repetition) INPUT: Sequence of items with (weight, value)

\((\omega_1, v_1), (\omega_2, v_2), \ldots, (\omega_n, v_n)\)

Maximum Total Weight \(= W\)

GOAL: Bundle of items with maximum total value

<table>
<thead>
<tr>
<th>ITEM</th>
<th>WEIGHT</th>
<th>VALUE</th>
</tr>
</thead>
<tbody>
<tr>
<td>A</td>
<td>15</td>
<td>43</td>
</tr>
<tr>
<td>B</td>
<td>6</td>
<td>18</td>
</tr>
<tr>
<td>C</td>
<td>7</td>
<td>19</td>
</tr>
<tr>
<td>D</td>
<td>8</td>
<td>23</td>
</tr>
<tr>
<td>E</td>
<td>5</td>
<td>14</td>
</tr>
<tr>
<td>F</td>
<td>4</td>
<td>9</td>
</tr>
</tbody>
</table>
DEFINE "SUBPROBLEMS:"

Imagine the optimal solution & break it up into smaller pieces.

\[ \text{Optimal Knapsack for } W = \{ \text{Optimal Knapsack for weight } W - w_A \} \]

Suppose \( K(w) = \text{optimal Knapsack for weight } w \) for each weight \( w = 1 \ldots W \).
Imagine:
In item \((w_n, v_n)\) part of it??

\[ K[w, i] = \text{optimal knapsack with total weight } \leq w \]
that uses only elements from \(q[1..i]\)

\[ K[W, n] = \text{Answer} \]
\[ K(\omega, F) = \text{optimal knapsack with total weight} \leq \omega \]

and using no forbidden elements.

\[ B_{1, \ldots, W^*} \cdot 2^n \]

\[ K(\omega, (n))^\uparrow \uparrow \]

\[ K(\omega, (n))^\uparrow \uparrow \]

\[ K(\omega, (n)) \]

\[ u \]

\[ u_n \]
**Recurrence Relation:**

\[ K[w,i] = \text{optimal knapsack with total weight } \leq w \text{ that uses only elements from } \{1..i\} \]

\[ K[w,i] = \max \begin{cases} 
\text{contains } i \text{th element} & V_i + K[w-w_i, i-1] \\
\text{does not contain } i \text{th element} & K[w, i-1]
\end{cases} \]
SUBPROBLEMS: (OA of subproblems)

\[ w = 0 \text{ to } W \]

\[ w_{j,i} = 0 \text{ to } W \]

\[ k(w_{j,i}) \]

\[ k(w_{j',i'}) \]

\[ i = 1 \text{ to } n \]

\[ j = 1 \text{ to } n \]
\[ D[i, j] = \min_k (D[i, k] + D(k, j)) \]
$$K[0, i] = 0 \quad \forall \ i \in \{1 \ldots n\}$$

$$K[\omega, 0] = 0 \quad \forall \ \omega \in \{1 \ldots W\}$$

for $i = 1$ to $n$

for $\omega = 1$ to $W$

$$K[\omega, i] = \max \begin{cases} 
(\omega < \omega) K[\omega - \omega_i, i-1] + v_i \\
K[\omega, i-1]
\end{cases}$$

return $K[W, n]$. 