# SOLO taxonomy: Problem-solving processes (5-6)

<table>
<thead>
<tr>
<th>SOLO LEVEL</th>
<th>One</th>
<th>Many</th>
<th>Relate</th>
<th>Extend</th>
</tr>
</thead>
<tbody>
<tr>
<td>SOLO VERB</td>
<td>Identify isolated skills</td>
<td>Describe and combine serial skills</td>
<td>Integrate skills</td>
<td>Evaluate skills</td>
</tr>
</tbody>
</table>

### DECLARATIVE KNOWLEDGE

**Knowing about (talking or writing about) the programming code**

Creating a digital solution using visual programming language

**Success criteria**

- I can **DEFINE** a problem identifying functional and data requirements
- I can **IDENTIFY** the use of isolated visual programming skills when programming
- For example, the use of:
  - an if/then statement
  - loops or repetition
  - user input

- I can **DESCRIBE** the use of isolated and combined visual programming skills when programming
- For example, the use of loops when:
  - incorporating repeat instructions
  - allowing for varied user input
  - selecting options (for example, in a quiz)

- AND I can **EXPLAIN** my programming choices – when programming a digital solution such as an animation, quiz, choose your own adventure story or controlling a robotic device
  - meeting its intended purpose
  - user input.

### FUNCTIONING KNOWLEDGE

**Knowing how to …**

Creating a digital solution using visual

- I can interpret an algorithm presented as a flow chart
- I can create an algorithm that I use to plan out a program for a digital solution.
- I can independently and confidently create a digital solution using a visual programming language
- AND I can seek and act on feedback to improve the effectiveness of my programming choices as I go.
**SOLO taxonomy: Problem-solving processes (5-6)**

<table>
<thead>
<tr>
<th>programming language</th>
<th>I can use a visual programming language <strong>IF</strong> I copy programming examples created by someone else</th>
<th>I can create a paper prototype of my design to show screen transitions</th>
<th>AND I can debug as I build (correct my own code)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Success criteria</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td><strong>Digital technologies</strong></td>
<td><strong>Way of thinking</strong></td>
<td><strong>Computational thinking</strong></td>
<td><strong>Computational thinking</strong></td>
</tr>
<tr>
<td>Design thinking</td>
<td>Computational thinking</td>
<td>Design thinking</td>
<td>AND I can debug as I build (correct my own code)</td>
</tr>
<tr>
<td>Computational thinking</td>
<td>Design thinking</td>
<td>Design thinking</td>
<td>Systems thinking</td>
</tr>
</tbody>
</table>

As learning progresses, it becomes more complex. SOLO stands for the Structure of the Observed Learning Outcome. It is a means of classifying learning outcomes in terms of their complexity. It can help differentiate a task to enable students to operate at their level and provide learning tasks that are progressively more challenging.

For more about SOLO Taxonomy refer to these websites

- [John Biggs Solo Taxonomy](#)
- [HookED: Solo Taxonomy](#)