Assessing Computational Thinking Skills

There are many ways in which you can monitor and assess your students' progress through a WeDo 2.0 project.

This section offers the following tools to help you in your assessments:

- Documentation pages
- Self-assessment statements
- Anecdotal record grid
- Observation rubrics grid

Student-led assessment

Documentation Pages

Each project will ask students to create documents to summarise their work.
To have a complete science report, it is essential that students:

- Document their work using various types of media
- Document every step of the process
- Take the time to organise and complete their document

It is most likely that the first document your students will complete will not be as good as the next one. You can support them by:

- Giving feedback and allowing them time to see where and how they can improve some parts of their document.
- Allowing them to share their documents with each other. By communicating their scientific findings, students will be engaged in the work of scientists.

Self-Assessment Statements

After each project, students should reflect on the work they have done. Use the following page to encourage reflection and set goals for the next project.

Student self-assessment rubric

Name:

Class:

Project:

				Directions: Circle the brick that shows how well you did. The bigger brick, the better you did.
I defined the question or problem.	○	○○	○○○	○○○○
I built a LEGO® model and programmed a solution.	○	○○	○○○	○○○○
I tested my solution and made improvements.	○	○○	○○○	○○○○
I documented and shared my ideas.	○	○○	○○○	○○○○

Project Reflection

One thing I did really well was:

One thing I want to improve on for next time is:

Teacher-led assessment

Developing students' science, engineering, and computational thinking skills requires time and feedback. Just as in the design cycle, in which students should understand that failure is part of the process, assessment should provide feedback in terms of what students did well and where they can improve. Problem-based learning is not about succeeding or failing. It is about being an active learner and continually building upon and testing ideas.

Giving feedback to students in order to help them develop their skills can be done in various ways. At each phase of the WeDo 2.0 projects, we have provided examples of rubrics that can be used by:

- Observing students' behaviour, reaction, and strategies
- Asking questions about their thought processes

As students often work in groups, you can give feedback both on a team level and on an individual level.

Anecdotal Record Grid

The anecdotal record grid lets you record any type of observation you believe is important for each student. Use the template on the next page to provide feedback to students as needed.

Anecdotal record grid

Name:

Class:

Project:

1. Emerging	2. Developing	3. Proficient	4. Accomplished

Notes:

Teacher-led assessment

Observation Rubrics

Examples of rubrics have been provided for every Guided Project. For every student, or every team, you can use the observation rubrics grid to:

- Evaluate student performance at each step of the process
- Provide constructive feedback to help the student progress

The observation rubrics provided in the Guided Projects can be adapted to fit your needs. The rubrics are based on these progressive stages:

1. Emerging
The student is at the beginning stages of development in terms of content knowledge, ability to understand and apply content, and/or demonstration of coherent thoughts about a given topic.

2. Developing
The student is able to present basic knowledge only (e.g., vocabulary), and cannot yet apply content knowledge or demonstrate comprehension of the concepts being presented.

3. Proficient
The student has concrete levels of comprehension of the content and concepts and can adequately demonstrate the topics, content, or concepts being taught. The ability to discuss and apply this knowledge outside the required assignment is lacking.

4. Accomplished
The student can take concepts and ideas to the next level, apply concepts to other situations, and synthesise apply, and extend knowledge to discussions that include extensions of ideas.

Suggestion

Use the observation rubrics grid on the next page to keep track of your students' progress.

Observation rubrics grid

Class:		Project:															
Students' Names		Explore	Create	Test	Share												
1																	
2																	
3																	
4																	
5																	
6																	
7																	
8																	
9																	
10																	
11																	
12																	
13																	
14																	
15																	

To be used with the rubrics described on the following page: (1) emerging, (2) developing, (3) proficient, (4) accomplished.

Assessing project phases - general rubrics

You can use these assessment rubrics to give general feedback on a scale of 1 to 4 at the end of each phase of a project.

Explore Phase

In the Explore phase, feedback should relate to whether or not the student is actively involved in the discussion by asking and answering questions, and their level of understanding of the problem.

1. The student is unable to provide answers to questions or adequately participate in discussions.
2. The student is able, with prompting, to provide answers to questions or adequately participate in discussions.
3. The student is able to provide adequate answers to questions and participate in class discussions.
4. The student is able to extend explanations in class discussions.

Test Phase

During the Test phase, make sure that the student works well on a team, justifies his/her best solution, and uses the information collected in the Explore phase.

1. The student is unable to work well on a team, justify solutions, and use information collected for further development.
2. The student is able to work on a team, collect and use information with guidance, or, with help, to justify solutions.
3. The student is able to work on a team and contribute to the team discussions, justify solutions, and collect and use information about the content.
4. The student can justify and discuss solutions that allow for the collection and use of information.

Share Phase

During the Share phase, make sure that the student is able to describe their solution using the right vocabulary and the right level of detail.

1. The student does not use evidence from his/her findings in connection with ideas shared during the presentation and does not follow established guidelines.
2. The student uses some evidence from his/her findings, but the justification is limited. Established guidelines are generally followed but may be lacking in one or more areas.
3. The student provides adequate evidence to justify his/her findings and follows established guidelines for presenting.
4. The student fully discusses his/her findings and thoroughly utilises appropriate evidence to justify his/her reasoning while following all established guidelines.

Assessing Computational Thinking Skills

Name:

Class:

Decomposition	1. Emerging	2. Developing	3. Proficient	4. Accomplished	Notes
	○	○ ○	○ ○ ○	○ ○ ○ ○	
Describe the problem in your own words.	The student is unable to describe the problem in their own words. □	The student is able, with prompting, to describe the problem in their own words. □	The student is able to describe the problem in their own words. □	The student is able to describe the problem in their own words and starts to decompose the problem into smaller parts. □	
Describe how you will know whether or not you have found a successful solution to the problem.	The student is unable to describe success criteria. □	The student is able, with prompting, to describe success criteria. □	The student is able to describe success criteria. □	The student is able to describe success criteria with a high level of detail. □	
Describe how you can break the problem down into smaller parts.	The student is unable to break down the problem. □	With prompting, the student is able to break down the problem into smaller parts. □	The student is able to break down the problem into smaller parts. □	The student is able to break down the problem into smaller parts and can describe the links between each of the parts. □	

Assessing Computational Thinking Skills

Name: Class:

Generalisation	1. Emerging	2. Developing	3. Proficient	4. Accomplished	Notes
	○	○ ○	○ ○ ○	○ ○ ○ ○	
Describe which program you have used from the Program Library (or elsewhere) and why.	The student is unable to describe which program has been used and why. □	The student is able to identify which program has been used. □	The student is able to describe which program has been used and why. □	The student is able to describe, in detail, which program has been used and what modifications have been made to it. □	
Observe how your students recognise patterns, or reuse concepts they have seen before.	The student is unable to recognise patterns, or reuse concepts seen before. □	With prompting, the student is able to recognise patterns, or reuse concepts seen before. □	The student is able to recognise patterns, or reuse concepts seen before. □	The student is able to recognise patterns, or reuse concepts of their own. □	

Assessing Computational Thinking Skills

Name:

Class:

Algorithmic Thinking	1. Emerging	2. Developing	3. Proficient	4. Accomplished	Notes
	○	○○	○ ○○	○○ ○○	
Describe the list of actions to program.	The student is unable to make a list of actions. □	With prompting, the student is able to make a list of actions. □	The student is able to make a list of actions. □	The student is able to make a detailed list of actions to help them develop their program. □	
Describe how you have programmed your solution.	The student is unable to describe the program. □	With prompting, the student is able to describe the program. □	The student is able to describe the program. □	The student is able to describe the program, providing extensive details about each component. □	
Describe the programming principles used in your solution (e.g., output, inputs, events, loops, etc.).	The student is unable to describe the programming principles used in their solution. □	With prompting, the student is able to describe the programming principles used in their solution. □	The student is able to describe the programming principles used in their solution. □	The student is able to describe, with extensive comprehension, the programming principles used in their solution. □	

Assessing Computational Thinking Skills

Name:

Class:

Evaluation	1. Emerging	2. Developing	3. Proficient	4. Accomplished	Notes
Describe what happened when you executed your program, and whether or not it was what you expected.	The student cannot describe what happened. □	With prompting, the student is able to describe what happened, and compare it to what was expected. □	The student is able to describe what happened, and compare it to what was expected. □	The student is able to describe what happened, compare it to what was expected, and is already finding solutions. □	
Describe how you have fixed the problems in your program.	The student cannot describe how they have fixed the problems. □	With prompting, the student can describe how they have fixed the problems. □	The student can describe how they have fixed the problems. □	The student can describe, in extensive detail, how they have fixed the problems. □	
Describe how your solution is linked to the problem.	The student is unable to describe how their solution is linked to the problem. □	With prompting, the student is able to describe how their solution is linked to the problem. □	The student is able to describe how their solution is linked to the problem. □	The student is able to describe, in extensive detail, how their solution is linked to the problem. □	
Describe how you have tried new ways of solving the problems throughout the project.	The student is unable to describe other ways tried throughout the project. □	The student is able, with prompting, to describe other ways tried throughout the project. □	The student is able to describe other ways tried throughout the project. □	The student is able to describe other ways tried throughout the project. and is able to describe why each option has not been considered. □	

Assessing Computational Thinking Skills

Name: Class:

Abstraction	1. Emerging	2. Developing	3. Proficient	4. Accomplished	Notes
	○	○○	○○○	○○○○	
Describe the most important part of your solution.	The student is not able to describe their solution. □	With prompting, the student is able to describe their solution. □	The student is able to describe their solution. □	The student is able to describe their solution, focusing on the most important part of the solution. □	
Describe the most important details of your solution.	The student is not able to provide any details about their solution. □	With prompting, the student is able to provide details about their solution. □	The student is able to discuss details of their solution, but some of the details are not essential. □	The student is able to discuss the most important details of their solution. □	
Describe how your solution met the initial criteria.	The student is unable to describe how their solution met the initial criteria. □	With prompting, the student is able to describe how their solution met the initial criteria. □	The student is able to describe how their solution met the initial criteria. □	The student is able to describe, with extraordinary clarity, how their solution met the initial criteria. □	