Teaching and Learning with Scratch

This 5-10 minute survey aims to surface opportunities for improving teaching and learning with Scratch, with a focus on grade 6-8 students. The responses will be reported in an aggregated and anonymized manner regardless of whether identifying information is provided. If you would like to receive a copy of the report when it is published, please provide your email address. Thank you for your assistance with this study. More information on Scratch is available here: https://scratch.mit.edu/.

* Required

1. **Email address** *

2. **Name (optional)**

3. **Email (optional)**

4. **How long have you been teaching with Scratch?** *
   Mark only one oval.
   - ◯ Less than 6 months
   - ◯ 6-12 months
   - ◯ 1-2 years
   - ◯ 2-4 years
   - ◯ More than 4 years

5. **Which subjects do you teach with Scratch?** *
   Check all that apply.
   - ☐ Science
   - ☐ Math
   - ☐ Language Arts
   - ☐ Computer Science
   - ☐ Social Studies
   - ☐ Applied Arts
   - ☐ Other:

6. **What percentage of the curriculum you teach involves Scratch?** *
Mark only one oval.

- 0%
- 1-5%
- 6-10%
- 11-15%
- 16-25%
- 26-50%
- 51-100%

7. **Which grades have you most frequently taught?** *
Mark only one oval.

- K-2
- 3-5
- 6-8
- 9-12
- Other: _____

Computational Thinking

Computational thinking is the thought process involved in formulating a problem and expressing its solution in a way a computer can carry it out. More information on computational thinking is available here: https://en.wikipedia.org/wiki/Computational_thinking.

To what extent do your students engage in the following activities when you teach with Scratch?

8. **Gathering appropriate information and selecting relevant information (data collection)** *
Mark only one oval.

	1	2	3	4	5	
Never						Always

9. **Making sense of data, finding patterns, drawing conclusions (data analysis)** *
Mark only one oval.

	1	2	3	4	5	
Never						Always

10. **Organizing and depicting data in appropriate graphs, charts, words, images, tables, etc. (data representation)** *
Mark only one oval.

	1	2	3	4	5	
Never						Always

11. **Breaking down tasks into smaller manageable parts and merging subtasks (problem decomposition)** *
Mark only one oval.

	1	2	3	4	5	
Never	◯	◯	◯	◯	◯	Always

12. **Planning and organizing sequences of steps taken to solve a problem (algorithms)** *
Mark only one oval.

	1	2	3	4	5	
Never	◯	◯	◯	◯	◯	Always

13. **Reducing complexity to define main idea, finding characteristics and creating models (abstraction)** *
Mark only one oval.

	1	2	3	4	5	
Never	◯	◯	◯	◯	◯	Always

14. **Using or creating simulations, for instance, for running experiments (simulation)** *
Mark only one oval.

	1	2	3	4	5	
Never	◯	◯	◯	◯	◯	Always

15. **Recognizing how technology can help us accomplish new tasks that would otherwise be too repetitive, infeasible, or difficult (automation)** *
Mark only one oval.

	1	2	3	4	5	
Never	◯	◯	◯	◯	◯	Always

16. **Organizing resources to simultaneously and cooperatively carry out tasks to reach a goal (parallelization)** *
Mark only one oval.

	1	2	3	4	5	
Never	◯	◯	◯	◯	◯	Always

17. **Please briefly describe one instance when you felt successful in including some of the above activities in your teaching practice.**

Direct Instruction

Direct instruction involves the explicit teaching of a skill-set through lecture, presentation, and demonstration. More information on direct instruction is available here: https://en.wikipedia.org/wiki/Direct_instruction.

18. **How frequently do your students ask for help when getting started with Scratch?** *
Mark only one oval.

	1	2	3	4	5	
Never	◯	◯	◯	◯	◯	Always

19. **How frequently do you supplement Scratch with direct instruction?** *
Mark only one oval.

	1	2	3	4	5	
Never	◯	◯	◯	◯	◯	Always

20. **How do you deliver direct instruction when teaching with Scratch?** *
Check all that apply.

- [ ] I don't deliver direct instruction when teaching with Scratch
- [ ] Concept explanations
- [ ] Demonstrations
- [ ] Written tutorials
- [ ] One-on-one tutoring
- [ ] Other:

21. **If Scratch included tools that facilitated introducing blocks throughout a learning progression, how willing would you be to use that functionality to support learning?** *
Mark only one oval.

	1	2	3	4	5	
Unwilling	◯	◯	◯	◯	◯	Willing

Parson's Programming Puzzles

Parson's Programming Puzzles (PPP) provide opportunities for practice with basic programming principles in an entertaining format. They are a family of code construction assignments in which lines of

code are given, and the task is to form the solution by sorting and selecting the correct code. Research indicates they might be a more efficient learning approach than fixing code with errors or writing equivalent code. More information on PPP is available here: http://crpit.com/confpapers/CRPITV52Parsons.pdf.

22. **How frequently do you assign PPP to students?** *
Mark only one oval.

	1	2	3	4	5	
Never	◯	◯	◯	◯	◯	Always

23. **If you assign PPP, how do you deliver them?** *
Check all that apply.

- [ ] I don't assign PPP
- [ ] Handcrafted paper assignments
- [ ] Hot Potatoes
- [ ] ViLLE
- [ ] CORT
- [ ] Other: _____

24. **If Scratch included effective PPP functionality, how willing would you be to assign these puzzles to students?** *
Mark only one oval.

	1	2	3	4	5	
Unwilling	◯	◯	◯	◯	◯	Willing

25. **If Scratch included effective PPP authoring functionality, how willing would you be to create these puzzles for your students?** *
Mark only one oval.

	1	2	3	4	5	
Unwilling	◯	◯	◯	◯	◯	Willing

Constructionist Video Games

Constructionist Video Games (CVG) are designed environments in which players construct personally meaningful artifacts in order to overcome conflicts or obstacles resulting in quantifiable outcomes. More information on CVG is available here: http://ccl.northwestern.edu/2016/WeintropHolbertHornWilensky2016.pdf.

26. **How frequently do your students engage in discovery-based learning in Scratch by creating their own projects?** *
Mark only one oval.

	1	2	3	4	5	
Never	◯	◯	◯	◯	◯	Always

27. **How frequently do students ask for project ideas?** *
Mark only one oval.

	1	2	3	4	5	
Never	◯	◯	◯	◯	◯	Always

28. **How frequently do you guide student discovery in Scratch with structured assignments?** *
Mark only one oval.

	1	2	3	4	5	
Never	◯	◯	◯	◯	◯	Always

29. **If Scratch included Constructionist Video Games intended to guide your students as they learn computational thinking, how willing would you be to assign those games to students?** *
Mark only one oval.

	1	2	3	4	5	
Unwilling	◯	◯	◯	◯	◯	Willing

30. **If Scratch included Constructionist Video Games creation, how willing would you be to design games for your students?** *
Mark only one oval.

	1	2	3	4	5	
Unwilling	◯	◯	◯	◯	◯	Willing

Intelligent Tutoring Systems

An intelligent tutoring system (ITS) provides immediate and customized instruction and feedback to students by a variety of delivery mechanisms such as just-in-time hints, on-demand information, and next-activity selection. More information on ITS is available here: https://en.wikipedia.org/wiki/Intelligent_tutoring_system.

31. **When your students are learning in Scratch, how frequently do they seek instructor guidance?** *
Mark only one oval.

	1	2	3	4	5	
Never	◯	◯	◯	◯	◯	Always

32. **How frequently does the rate of student questions exceed your capacity to provide guidance promptly to individuals?** *
Mark only one oval.

	1	2	3	4	5	
Never	◯	◯	◯	◯	◯	Always

33. **If Scratch included ITS functionality, how willing would you be to enable that functionality to complement your teaching?** *

Mark only one oval.

	1	2	3	4	5	
Unwilling	◯	◯	◯	◯	◯	Willing

34. **If you had the opportunity to request one new feature intended to facilitate teaching and learning with Scratch, what would it be?**

35. **What other software do you use that is similar to Scratch?**

36. **Are there any questions and/or terms in this survey that are difficult to understand? Your candid feedback will help inform the interpretation of the results, and help guide the design of future studies.**

☐ Send me a copy of my responses.