A KIBO Coding Curriculum for Readers

Integrated with Foundational Literacy Topics

Using the KIBO robotics kit and Coding as Literacy (CAL) approach developed by

DevTech Research Group
Eliot-Pearson Dept. of Child Study and Human Development
Tufts University

Table of Contents

INTRODUCTION

LESSONS

APPENDICES & REFERENCES

Introduction

CODING AS LITERACY (CAL) APPROACH

This curriculum introduces powerful ideas from computer science, specifically programming with KIBO robotics, to young children in a structured, developmentally appropriate way. The **Coding as Literacy (CAL)** approach, developed by Prof. Marina Umaschi Bers and members of her DevTech Research Group at Tufts University, puts computer science ideas into direct conversation with powerful ideas from literacy. The starting assumption of the CAL curriculum is that both computer science and literacy can enhance one another. Instruction in both can be leveraged in service of the other. Both can support learners in developing new ways of thinking about themselves and the world.

Thinking involves the ability to make sense of, interpret, represent, model, predict, and invent our experiences in the world. Thus, as educators, we must give children one of the most powerful tools for thinking: language. The term **language** refers here to a system of communication, natural or artificial, composed of a formal limited system of signs, governed by syntactic and grammatical combinatory rules, that serves to communicate meaning by encoding and decoding information. Today, we have the opportunity to not only teach children how to think by using natural languages, such as English, but also by learning artificial languages—programming languages such as the one used by KIBO robots.

The achievement of literacy in a natural language involves a progression of skills beginning with the ability to understand spoken words, followed by the capacity to code and decode written words, and culminating in the deep understanding, interpretation, and production of text. The ultimate goal of literacy is not only for children to master the syntax and grammar, the orthography and morphology, but also the semantics and pragmatics, the meanings and uses of words, sentences and genres. A literate person knows that reading and writing are tools for meaning making and, ultimately, tools of power because they support new ways of thinking.

The CAL approach proposes that programming, as a literacy of the 21st century, engages new ways of thinking and new ways of communicating and expressing ideas, as well as new ways of problem solving and working with others. CAL understands the process of coding as a semiotic act, a meaning making activity that engages children in both developing computational thinking, as well as promoting personal expression, communication, and interpretation. This understanding shapes this curriculum and our strategies for teaching coding.

The curriculum is organized around **powerful ideas** from both computer science and literacy. The term powerful idea refers to a central concept or skills within a discipline that is simultaneously personally useful, inherently interconnected with other disciplines, and has roots in intuitive knowledge that a child has internalized over a long period of time. The **powerful ideas from computer science** addressed in this curriculum include: algorithms, design process, representation, debugging, control structures, modularity, and hardware/software. The **powerful ideas from literacy** that will be placed in conversation with these powerful ideas from computer science are: the writing process, recalling, summarizing and sequencing, using illustrative and descriptive language, recognizing literary devices such as repetition and foreshadowing, and using reading strategies such as predicting, summarizing, and evaluating.

The CAL approach allows students to make connections between coding and literacy and use the two platforms to express their thoughts and ideas. These powerful ideas of literacy and computer science are explored in the context of a curriculum that draws on the well-known children's book *Where the Wild Things Are* by Maurice Sendak, which is about a young boy named Max who makes mischief at home and then sails to the land where the wild things are.

Each lesson contains a variety of activities to introduce children to programming and literacy skills and concepts. Lessons are aligned to academic frameworks of Common Core, as well as Virginia Public Schools, as in 2017, Virginia became the first state in the US to formally mandate K-12 computer science standards. Teachers are encouraged to use this curriculum as a guiding resource and to adapt lessons and activities to their needs of their students. Activities in this curriculum include:

- Warm up games to playfully introduce or reinforce concepts
- Design challenges to introduce the powerful ideas from computer science
- Writing activities to introduce the powerful ideas from literacy
- Work individually or in pairs on designing and creating projects
- Technology circles to share and reflect on activities
- Free-explorations to allow students to tinker and expand their skills

The culmination of the unit is an open-ended project to share with family and friends. Just as young children can read age-appropriate books, computer programming can be made accessible by providing young children with appropriate tools such as KIBO.

PACING

This 14-hour curriculum unit is designed to take place over the course of a few months with one or two sessions per week (i.e. 1-2 hours each week for 2-3 consecutive months). This curriculum provides suggested time allotments, but they should be adapted to suit the needs of each classroom.

To supplement the structured challenges, free-exploration is allotted throughout the curriculum. These open-ended sessions are vital for children to fully understand the complex ideas behind their robotic creations and programs. The free-exploration sessions also serve as a time for teachers to observe students' progress and understandings. These sessions are as important for learning as the lessons themselves! In planning and adjusting the timeframe of this curriculum, free-exploration sessions should not be left by the wayside. Free-exploration provides opportunities for playing with materials and ideas and will help build a solid foundation.

Table 1: Pacing Guide

Lesson	**Activities**
Lesson 1: Foundations	• What is an Engineer? (20 min) • Engineers and Writers (10 min) • Think Like an Engineer (10 min) • How to Make a Pizza (20 min)
Lesson 2: Technological Tools - Robots	• Robot Corners (15 min) • Characteristics of Robots (10 min) • Tools of Communication (10 min) • Human Language vs. Code Language (10 min) • Progammer Says (15 min)

Lesson	Activities
Lesson 3: Sequencing	• Where the Wild Things Are (15 min) • Writing Reflection (20 min) • Program the Teacher with KIBO Blocks (10 min) • Meet the KIBO Robot (10 min) • Clean up (5 min)
Lesson 4: Programming	• Dance the Hokey Pokey (5 min) • Program the Hokey Pokey (20 min) • Hokey-Pokey Reflection (10 min) • Share Creations (5 min) • Clean Up (5 min) • KIBO Mastery Challenge A (15 min)
Lesson 5: Debugging	• Tell a Story (20 min) • Why is KIBO Confused? (15 min) • Free Play (10 min) • Debugging Reflection (10 min) • Clean Up (5 min)
Lesson 6: Cause and Effect - Level 1	• What did Max Sense (15 min) • KIBO Sound Sensor (5 min) • Shape Shifting (10 min) • KIBO Sound Recorder (5 min) • Free Play (10 min) • Clean Up (5 min) • KIBO Mastery Challenge B (10 min)
Lesson 7: Cause and Effect - Level 2	• Sing “If You’re Wild and You Know It” (5 min) • Program "If You're Wild and You Know It" (30 min) • Project Reflection (10 min) • Share Creations (10 min) • Clean Up (5 min)
Lesson 8: Repeat Loops - Level 1	• Repetition in Stories and Songs (15 min) • Toothbrush Exercise (15 min) • KIBO Repeat with Numbers (15 min) • Clean Up (5 min) • KIBO Mastery Challenge C (10 min)
Lesson 9: Repeat Loops - Level 2	• My Five Senses (20 min) • KIBO Repeat with Light Sensors (10 min) • Free Play with Repeats (25 min) • Clean Up (5 min)

Lesson	Activities
Lesson 10: Repeat and If Statements	• Write an Alternative Story (20 min) • KIBO Repeat with Distance Sensor (15 min) • Free Play (20 min) • Optional Extension Activity: KIBO If Statements • Optional Extension Activity: KIBO Nested Statements • Clean Up (5 min)
Lesson 11: Final Project - Writing the Wild Rumpus Composition	• Wild Rumpus Composition (20 min) • Writing vs. Coding (10 min) • Begin Designing Projects (20 min) • Technology Circle (10 min)

Lesson 12: Final Project - Coding the Wild Rumpus (3 Day Plan)	• Day 1
	  • Plan and Create (20 min)
	  • Coding the Wild Rumpus (20 min)
	  • Test and Improve (10 min)
	  • Clean Up and Prep for Next Day (10 min)
	• Day 2
	  • Revise Projects (15 min)
	  • Peer Feedback (10 min)
	  • Collaboration Web (10 min)
	  • Decorating KIBO (20 min)
	  • Clean Up (5 min)
	• Day 3
	  • Prepare for Sharing (5 min)
	  • Share Creations (20 min)
	  • Clean Up (5 min)
	  • Wild Rumpus Reflection (15 min)
	  • KIBO Mastery Challenge D (15 min)

MATERIALS

The robotics kit referred to in this curriculum is the KIBO robotics kit, developed by the DevTech Research Group at Tufts University and made commercially available through KinderLab Robotics, Inc. (www.kinderlabrobotics.com). This curriculum uses the KIBO 21 kit, which includes the following:

KIBO robot with wheels and motors

Input/output modules (distance, sound, and light sensors, lightbulb, sound recorder)

Expression module

Stage art platform

Rotating art stage with motor

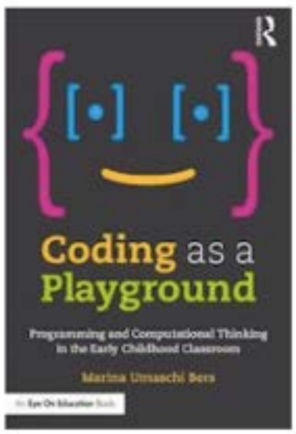
21 programming blocks

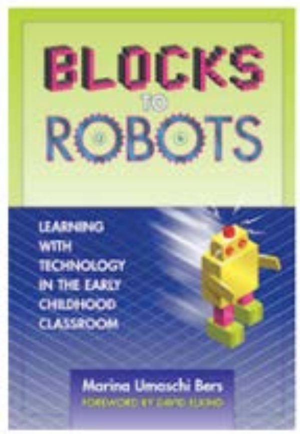
12 parameter cards

Other materials used in the curriculum are inexpensive crafts and recycled materials. The use of crafts and recycled materials, a practice already common in other domains of early childhood education, lets children build with a range of materials with which they are already comfortable with. There are many supplemental materials such as the KIBO Says cards and Activity Guide Cards that can be purchased through KinderLab Robotics (www.kinderlabrobotics.com). See Appendix A for the full list of materials for this curriculum.

PEDAGOGICAL FRAMEWORK: POSITIVE TECHNOLOGICAL DEVELOPMENT and DIALOGIC INSTRUCTION

The theoretical foundation of this curriculum, called **Positive Technological Development** (PTD), was developed by Prof. Marina Umaschi Bers and can be found in her books: *Blocks to Robotics: Learning with Technology in the Early Childhood Classroom* (Bers, 2008), *Designing Digital Experiences for Positive Youth Development: From Playpen to Playground* (Bers, 2012), and *Coding as a Playground: Programming and Computational Thinking in the Early Childhood Classroom* (Bers, 2018). More information is included in the References section at the end of this curriculum.





The PTD framework guides the development, implementation and evaluation of educational programs that use new technologies to promote learning as an aspect of positive youth development. The PTD framework is a natural extension of the computer literacy and the technological fluency movements that have influenced the world of education but adds psychosocial and ethical components to the cognitive ones. From a theoretical perspective, PTD is an interdisciplinary approach that integrates ideas from the fields of computer-mediated communication, computer-supported collaborative learning, and the Constructionist theory of learning developed by Seymour Papert (1993) and views them in light of research in applied development science and positive youth development.

As a theoretical framework, PTD proposes six positive behaviors (six C's) that should be supported by educational programs that use new educational technologies, such as KIBO robotics. These are: **content creation, creativity, communication, collaboration, community building, and choices of conduct**. The six C's of PTD are highlighted in the activities throughout the curriculum with their respective icons:

CONTENT CREATION by designing a KIBO robot and programming its behaviors. The engineering design process of building and the computational thinking involved in programming foster competence in computer literacy and technological fluency. The use of Design Journals document for the children themselves, as well as for teachers and parents, their own thinking, their learning trajectories and the project's evolution over time.

CREATIVITY by making and programming personally meaningful projects, problem solving in creative playful ways and integrating different media such as robotics, motors, sensors, recyclable materials, arts and crafts, and a tangible programming language. Final KIBO projects that represent a theme found in the overall early childhood curriculum are a wonderful way to engage children in the creative process of learning.

COLLABORATION by engaging children in a learning environment that promotes working in teams, sharing resources and caring about each other while working with their KIBO robots. Collaboration is defined here as getting or giving help with a project, programming together, lending or borrowing materials, or working together on a common task. Teachers are encouraged to use the Job Cards (see Appendix A) to allow children to take turns and assume multiple roles while working with KIBO. On their final KIBO projects, children have the option to create a collaboration web (see Appendix D for an example). Each child receives a printout with their photograph in the center of the page and the names and photographs of all the other children in the class arranged in a circle surrounding the central photo. Throughout the activity, with the teacher's prompting, each child draws a line from their own photo to the photos of the other children with whom they have collaborated. Children then write or draw "thank you cards" to the children with whom they have collaborated the most.

COMMUNICATION through mechanisms that promote a sense of connection between peers or with adults. For example, technology circles, when children stop their work, put their projects on the table or floor, and share their learning process. Technology circles present a good opportunity for problem solving as a community. Some teachers invite all the children to sit together in the rug area for this. It can also be helpful to make a "Robot Parking Lot" for all the robots to go while they are not being worked on, so children have empty hands and can focus at the technology circles. Each classroom will have its own routines and expectations around group discussions and circle times, so teachers are encouraged to adapt what already works in their class for the technology circles in this curriculum.

COMMUNITY BUILDING through scaffolded opportunities to form a learning community that promotes contribution of ideas. Final projects done by children are shared with the community via an open house, demo day, or exhibition. These open houses provide authentic opportunities for children to share and celebrate the process and tangible products of their learning with family and friends. Each child is given the opportunity not only to run their robot, but to play the role of teacher as they explain to their family how they built, programmed, and worked through problems.

CHOICES OF CONDUCT which provide children with the opportunity to experiment with "what if" questions and potential consequences, and to provoke examination of values and exploration of character traits while working with robotics. As a program developed following the PTD approach, the focus on learning about robotics is as important as helping children develop an inner compass to guide their actions in a just and responsible way.

In alignment with the Positive Technological Development (PTD) framework, this curriculum approaches literacy from the perspective of dialogic instruction. **Dialogic instruction** is a theory of learning (and teaching) premised on the belief that students engage with literacy instruction best when there are opportunities for them to engage in authentic, open-ended interpretation of texts. If a student does not have a voice, a position, or an evaluation of the text, then what good are literary skills? Only when she needs these tools for her own purpose, to help her achieve her own interpretation, and to convince others of it, will she have a reason and motivation (beyond getting a good grade) to acquire the tools being taught. This curriculum, in adherence with the theory of dialogic instruction, strives to place the student in the position of interpreter, with opportunities for authentic, open-ended interpretation of texts. This aligns with the curriculum's approach to coding where students are given opportunities for open-ended coding tasks that encourage them to explore their own expressive ideas.

CLASSROOM MANAGEMENT

Teaching robotics and programming in an early childhood setting requires careful planning and ongoing adjustments when it comes to classroom management issues. These issues are not new to the early childhood teacher, but they may play out differently during robotics activities because of the novelty and behavior of the materials themselves. Issues and solutions other than those described here may arise from classroom to classroom; teachers should find what works in their particular circumstances. In general, provide and teach a clear structure and set of expectations for using materials and for the routines of each part of the lessons (technology circles, clean up time, etc.). Make sure the students understand the goal(s) of each activity. Posters and visual aids can facilitate children's attempts to answer their own questions and recall new information. For example, teachers can use the mnemonic "KIBO" to introduce norms for playing with the KIBO robotics kit: **K**udos to…, **I** respect you, you respect me, **B**odies are safe, and **O**ops! Let's try it again.

GROUP SIZES

The curriculum refers to whole-group versus pair or individual work. In fact, some classrooms may benefit from other groupings. Whether individual work is feasible depends on the availability of supplies, which may be limited for a number of reasons. However, an effort should be made to allow students to work in as small groups as possible, even individually. At the same time, the curriculum includes numerous opportunities to promote conversations which are enriched by multiple voices, viewpoints, and experiences. Some classes may be able to have these discussions as a whole group. Other classes may want to break up into smaller groups to allow more children the opportunity to speak and to maintain focus. Some classes structure robotics time to fit into a "center time" in the schedule, in which students rotate through small stations around the room with different activities at each location. This format gives students more access to teachers when they have questions and lets teachers tailor instruction and feedback as well as assess each students' progress more easily than during whole-group work. It is important to find a structure and group size for each of the different activities (instruction, discussions, work on the challenges, and the final project) that meet the needs of the students and teachers in the class.

MANAGING MATERIALS

Classroom-scale robotics projects require a lot of parts and materials, and the question of how to manage them brings up several key issues that can support or hinder the success of the unit.

The first issue is accessibility of materials. The recommended ratio is 1 KIBO per 2-4 children, so that all children have a chance to interact with KIBO. Some teachers may choose to give a complete kit of materials to each child, pair, or table of several children. Children may label the kit with their name(s) and use the same kit for the duration of the curriculum. Other teachers may choose to take apart the kits and have materials sorted by type and place all the materials in a central location. Since different projects require different robotic and programming elements, this setup may allow children to take only what they need and leave other parts for children who need them. A word of caution, however: If materials are set up centrally, they must be readily visible and accessible, so children don't forget what is available to them or find it too much of a hassle to get what they need. Regardless, it is important to find a clearly visible place to set up materials for demonstrations, posters or visual aids to display for reference, and for robotics and programming materials for each lesson. To facilitate teamwork and equal participation, Appendix A includes examples of Job Cards that can be distributed to the children, which will assign specific roles to each child when working with KIBO.

The second issue is usability. In some cases, children's desks or tables do not provide enough space to build a robot and program it. Care must be taken to ensure that children have enough space to use the materials available to them. If this is not the case, they may tend towards choosing materials that fit the space but not their robotics or programming goal.

Teachers should carefully consider how to address these issues surrounding materials in a way that makes sense for their class's space, routines, and culture. Then, it is crucial to set expectations for how to use and treat materials. These issues are important not only in making the curriculum logistically easier to implement, but also because, as described in the Reggio Emilia tradition, the environment can act as the "third teacher" (Darragh, 2006).

ALIGNMENT OF ACADEMIC FRAMEWORKS

This curriculum is designed for young children and covers many foundational computer science and engineering skills. These academic frameworks are taught through a series of powerful ideas: algorithms, modularity, control structures, representation, hardware/software, design process, and debugging. Each powerful idea has activities and materials (in this case, the activities are tailored to fit the theme of *Where the Wild Things Are*) that encourage mastery of the powerful ideas from computational thinking (CT) and matches them with corresponding powerful ideas from literacy. This curriculum contains activities that specifically address the following literacy concepts and skills: the writing process, recalling, summarizing and sequencing, using foreshadowing, and using reading strategies such as predicting, summarizing, and evaluating.

Each lesson in this curriculum unit is aligned with standards from the **Common Core English Language Arts (ELA)/Literacy Framework**. The Common Core framework is "a set of standards that were created to ensure that all students graduate from high school with the skills and knowledge necessary to succeed in college, career, and life, regardless of where they live" (National Governors Association Center for Best Practices & Council of Chief State School Officers, 2010). Because Virginia is the first state to formally mandate K-12 computer science education, this curriculum is aligned with the **Virginia Department of Education's Standards of Learning for English and Standards of Learning for Computer Science** (Virginia Department of Education, 2017). Lessons in this curriculum are also aligned with nationally recognized computer science frameworks, including the **ISTE Standards for Students** (2017), **K–12 Computer Science Framework** (2016) and the **Massachusetts Digital Literacy and Computer Science (DLCS) Curriculum Framework** (2016).

Table 2: Alignment of Standards

	Powerful Ideas of Computational Thinking (CT) and Literacy Embedded in Each Lesson	Common Core ELA/ Literacy Framework (Grade 1)	Virginia English Standards of Learning (Grade 1)	Virginia Computer Science Standards of Learning (Grade 1)
1: Foundations	***CT***: *Design Process* ***Literacy***: *Writing Process*	**CCSS.ELA-LITERACY.W.1.5** With guidance and support from adults, focus on a topic, respond to questions and suggestions from peers, and add details to strengthen writing as needed.	**Writing 1.12e** The student will revise by adding descriptive words when writing about people, places, things, and events. **Writing 1.13** The student will edit writing for capitalization, punctuation, and spelling.	**Algorithms and Programming 1.6** The student will acknowledge that materials are created by others.

2: Technological Tools - Robots	***CT***: *Hardware/ Software, Representation* ***Literacy***: *Tools of Communication*	**CCSS.ELA-LITERACY.W.1.6** With guidance and support from adults, use a variety of digital tools to produce and publish writing, including in collaboration with peers.	**Communication 1.2** The student will demonstrate growth in oral early literacy skills. **Communication 1.2a** The student will listen and respond to a variety of print and media materials.	**Algorithms and Programming 1.6** The student will acknowledge that materials are created by others. **Cybersecurity 1.9** The student will describe what is allowed and what is not allowed at school associated with the use of technology
3: Sequencing	***CT***: *Hardware/ Software, Algorithms, Representation* ***Literacy***: *Summarizing/ Retelling the Sequence of a Story, Descriptive Language in Writing*	**CCSS.ELA-LITERACY.W.1.3** Write narratives in which they recount two or more appropriately sequenced events, include some details regarding what happened, use temporal words to signal event order, and provide some sense of closure. **CCSS.ELA-LITERACY.RL.1.3** Describe characters, settings, and major events of a story, using key details. **CCSS.ELA-LITERACY.RF.1.3.A** Know spelling-sound correspondences for common consonant digraphs.	**Reading 1.5** The student will apply phonetic principles to read and spell. a) Use initial and final consonants to decode and spell one-syllable words. b) Use two-letter consonant blends to decode and spell one-syllable words. c) Use consonant digraphs to decode and spell one-syllable words. **Reading 1.9** The student will read and demonstrate comprehension of a variety of fictional texts. **Reading 1.9g** The student will retell stories and events using the beginning, middle, and end in sequential order.	**Algorithms and Programming 1.2** The student will construct programs to accomplish tasks as a means of creative expression using a block based programming language or unplugged activities, either independently or collaboratively including a) sequencing, ordinal numbers; and b) simple loops (patterns and repetition) **Algorithms and Programming 1.4** The student will plan and create a design document to illustrate thoughts, ideas and stories in sequential (step-by-step) manner (e.g. sequential graphic organizer)
	CT: *Algorithms, Design Process* ***Literacy***: *Descriptive Language in Writing*	**CCSS.ELA-LITERACY.W.1.2** Write informative/ explanatory texts in which they name a topic, supply some facts about the topic, and provide some sense of closure.	**Communication 1.1g** The student will ask and respond to question to seek help, get information, or clarify information. **Communication 1.1j** The student will express ideas orally in complete sentences.	**Algorithms and Programming 1.2** The student will construct programs to accomplish tasks as a means of creative expression using a block based programming language or unplugged activities, either independently or collaboratively including a) sequencing, ordinal numbers; and b) simple loops (patterns and repetition)

4. Programming		**CCSS.ELA-LITERACY.W.1.3** Write narratives in which they recount two or more appropriately sequenced events, include some details regarding what happened, use temporal words to signal event order, and provide some sense of closure.	**Communication 1.2b** The student will tell and retell stories and events in sequential order.	**Algorithms and Programming 1.3** The student will analyze and debug (correct and improve) an algorithm that includes sequencing. **Algorithms and Programming 1.4** The student will plan and create a design document to illustrate thoughts, ideas and stories in sequential (step-by-step) manner (e.g. sequential graphic organizer) **1.8** The student will identify, using accurate terminology, simple hardware and software problems that may occur during use (e.g., app or program is not working as expected, no sound is coming from device, the device won't turn on).
5. Debugging	***CT***: *Debugging* ***Literacy***: *Editing, Awareness of Audience*	**CCSS.ELA-LITERACY.RL.1.4** Identify words and phrases in stories or poems that suggest feelings or appeal to the senses.	**Writing 1.12e** Students will revise by adding descriptive words when writing about people, places, things and events. **Communication 1.1** The student will develop oral communication skills. **Communication 1.1l** The student will increase listening and speaking vocabularies	**Algorithms and Programming 1.2** The student will construct programs to accomplish tasks as a means of creative expression using a block based programming language or unplugged activities, either independently or collaboratively including a) sequencing, ordinal numbers; and b) simple loops (patterns and repetition) **Algorithms and Programming 1.4** The student will plan and create a design document to illustrate thoughts, ideas and stories in sequential (step-by-step) manner (e.g. sequential graphic organizer) **Algorithms and Programming 1.5** The student will categorize a group of items based on one or two attributes or the actions of each item, with or without a computing device.
6: Cause and Effect - Level 1	***CT***: *Control Structures, Representation, Sensors* ***Literacy***: *Spelling-Sound Correspondence*	**CCSS.ELA-LITERACY.RF.1.3** Know and apply grade-level phonics and word analysis skills in decoding words.	**Communication 1.2** The student will demonstrate growth in oral early literacy skills. **Communication 1.2c** The student will participate in a variety of oral language activities, including choral speaking and recitation.	**Algorithms and Programming 1.2** The student will construct programs to accomplish tasks as a means of creative expression using a block based programming language or unplugged activities, either independently or collaboratively including a) sequencing, ordinal numbers; and b) simple loops (patterns and repetition).

			Communication 1.2d The student will participate in creative dramatics. **Reading 1.3** The student will orally identify, produce, and manipulate various phonemes within words to develop phonological and phonemic awareness.	**Algorithms and Programming 1.4** The student will plan and create a design document to illustrate thoughts, ideas and stories in sequential (step-by-step) manner (e.g. sequential graphic organizer)
7: Cause and Effect - Level 2	***CT****: Algorithms, Modularity, Representation* ***Literacy****: Descriptive Language in Writing*	**CCSS.ELA-LITERACY.RF.1.3.E** Decode two-syllable words following basic patterns by breaking the words into syllables. **CCSS.ELA-LITERACY.RL.1.4** Determine or clarify the meaning of unknown and multiple-meaning words and phrases based on grade 1 reading and content, choosing flexibly from an array of strategies	**Reading 1.7c** Students will ask for the meaning of unknown words and make connections to familiar words. **Reading 1.9** The student will read and demonstrate comprehension of a variety of fictional texts.	**Algorithms and Programming 1.1** The student will construct algorithms (the set of step-by-step instructions) either independently or collaboratively, including a) sequencing and ordinal numbers and b) simple loops (patterns and repetition) **Algorithms and Programming 1.2** The student will construct programs to accomplish tasks as a means of creative expression using a block based programming language or unplugged activities, either independently or collaboratively including a) sequencing, ordinal numbers; and b) simple loops (patterns and repetition) **Algorithms and Programming 1.4** The student will plan and create a design document to illustrate thoughts, ideas and stories in sequential (step-by-step) manner (e.g. sequential graphic organizer)
8: Repeat Loops - Level 1	***CT****: Control Structure, Modularity* ***Literacy****: Repetition as a Literacy Device, Repetition in Word Forms*	**CCSS.ELA-LITERACY.RL.1.4** Describe people, places, things and events with relevant details, expressing ideas and feelings clearly.	**Reading 1.9** The student will read and demonstrate comprehension of a variety of fictional texts. **Reading 1.9i** The student will read and reread familiar stories and poems with fluency, accuracy and meaningful expression	**Algorithms and Programming 1.1** The student will construct algorithms (the set of step-by-step instructions) either independently or collaboratively, including a) sequencing and ordinal numbers and b) simple loops (patterns and repetition) **Algorithms and Programming 1.2** The student will construct programs to accomplish tasks as a means of creative expression using a block based programming language or unplugged activities, either independently or collaboratively including a) sequencing, ordinal numbers; and b) simple loops (patterns and repetition)

9: Repeat Loops - Level 2	***CT**: Control Structures, Debugging, Sensors* ***Literacy**: Descriptive Language, Perspectives in Narrative*		**Communication 1.1** The student will develop oral communication skills. **Communication 1.1h** The student will restate and follow simple two-step directions. **Communication 1.1i** The student will give simple two-step oral directions.	**Algorithms and Programming 1.1** The student will construct algorithms (the set of step-by-step instructions) either independently or collaboratively, including a) sequencing and ordinal numbers and b) simple loops (patterns and repetition) **Algorithms and Programming 1.2** The student will construct programs to accomplish tasks as a means of creative expression using a block based programming language or unplugged activities, either independently or collaboratively including a) sequencing, ordinal numbers; and b) simple loops (patterns and repetition)
10: Repeat and If Statements	***CT**: Control Structures, Debugging, Sensors* ***Literacy**: Identify Conflict and Resolution, Making Predictions*	**CCSS.ELA-LITERACY.RI.1.3** Describe the connection between two individuals events, ideas, or pieces of information in a text.	**Reading 1.9** The student will read and demonstrate comprehension of a variety of fictional texts. **Reading 1.9d** The student will make and confirm predictions. **Reading 1.9e** The student will ask and answer who, what, when, where, why, and how questions about what is read.	**Algorithms and Programming 1.1** The student will construct algorithms (the set of step-by-step instructions) either independently or collaboratively, including a) sequencing and ordinal numbers and b) simple loops (patterns and repetition) **Algorithms and Programming 1.2** The student will construct programs to accomplish tasks as a means of creative expression using a block based programming language or unplugged activities, either independently or collaboratively including a) sequencing, ordinal numbers; and b) simple loops (patterns and repetition)
11: Final Project - Writing the Wild Rumpus Composition	***CT**: Design Process* ***Literacy**: Writing Process*	**CCSS.ELA-LITERACY.W.1.5** With guidance and support from adults and peers, focus on a topic, respond to questions and suggestions from peers, and add details to strengthen writing as needed.	**Writing 1.12** The student will write in a variety of forms to include narrative, descriptive and opinion. **Writing 1.12b** The student will use prewriting activities to generate ideas. **Writing 1.12d** The student will organize writing to suit purpose. **Writing 1.12e** The student will revise by adding descriptive words when writing about people, places, things, and events.	**Algorithms and Programming 1.1** The student will construct algorithms (the set of step-by-step instructions) either independently or collaboratively, including a) sequencing and ordinal numbers and b) simple loops (patterns and repetition) **Algorithms and Programming 1.2** The student will construct programs to accomplish tasks as a means of creative expression using a block based programming language or unplugged activities, either independently or collaboratively including a) sequencing, ordinal numbers; and b) simple loops (patterns and repetition) **Algorithms and Programming 1.3** The student will analyze and debug (correct and improve) an algorithm that includes sequencing.

				Algorithms and Programming 1.4 The student will plan and create a design document to illustrate thoughts, ideas and stories in sequential (step-by-step) manner (e.g. sequential graphic organizer)
12. Final Project - Coding the Wild Rumpus	***CT***: *Design Process* ***Literacy***: *Writing Process*	**CCSS.ELA-LITERACY.W.1.5** With guidance and support from adults and peers, focus on a topic, respond to questions and suggestions from peers, and add details to strengthen writing as needed.	**Writing 1.12** The student will write in a variety of forms to include narrative, descriptive and opinion. **Writing 1.12b** The student will use prewriting activities to generate ideas. **Writing 1.12d** The student will organize writing to suit purpose. **Writing 1.12e** The student will revise by adding descriptive words when writing about people, places, things, and events. **Writing 1.13** The student will edit writing for capitalization, punctuation, and spelling. **Reading 1.9** The student will read and demonstrate comprehension of a variety of fictional texts. **Reading 1.9f** The student will identify characters, setting, and important events.	**Algorithms and Programming 1.1** The student will construct algorithms (the set of step-by-step instructions) either independently or collaboratively, including a) sequencing and ordinal numbers and b) simple loops (patterns and repetition) **Algorithms and Programming 1.2** The student will construct programs to accomplish tasks as a means of creative expression using a block based programming language or unplugged activities, either independently or collaboratively including a) sequencing, ordinal numbers; and b) simple loops (patterns and repetition) **Algorithms and Programming 1.3** The student will analyze and debug (correct and improve) an algorithm that includes sequencing. **Algorithms and Programming 1.4** The student will plan and create a design document to illustrate thoughts, ideas and stories in sequential (step-by-step) manner (e.g. sequential graphic organizer)

Lesson 1: Foundations

Powerful Idea From Computer Science:
Design Process

Powerful Idea From Literacy:
Writing Process

OVERVIEW

Students will learn about the Design Process and the Writing Process and understand how both processes are similar in nature but serve different purposes. Activities in this lesson encourage students to think and act like engineers and writers.

PURPOSE

While this lesson does not involve using the KIBO robotics kit, the activities set up an important foundation for how students engage in key computer science and literacy skills, such as brainstorming ideas, planning out a project, reviewing and revising ideas, and sharing ideas with peers.

ACTIVITIES

- What is an Engineer? (20 min)
- Engineers and Writers (10 min)
- Think Like an Engineer (10 min)
- How to Make a Pizza (20 min)

STUDENTS WILL BE ABLE TO...

- Define engineer and understand that there are different types of engineers
- Compare and contrast the Design Process and Writing Process
- Use the Design and Writing Processes to write a set of instructions for making something

PREPARATION FOR TEACHERS

- ☐ Read through the Activity Guide
- ☐ Review the Lesson Slides
- ☐ Gather all materials for the lesson

MATERIALS

FOR THE TEACHER:

- 8-10 pictures of naturally occurring and man-made objects*
- Anchor chart of Design Process*
- Anchor chart of Writing Process*
- How-to book checklist

FOR STUDENTS:

- Design Journal (see Appendix C for example)

*See Lesson Slides and Appendix A for examples

VOCABULARY

- Author – a creator of a product in writing (i.e., book, article, program, etc.)
- Cycle — something that moves in a circle (i.e. the seasons, a baseball field (compare to a football field that goes forward and backwards) the Design Process, the Writing Process)
- Design — a plan for a building or invention
- Digital artifact – digital content made by a human with intent or skill
- Engineer — someone who invents or improves things
- Illustrator – a creator of a picture, graphic or visual art product

WHAT IS AN ENGINEER? (20 min)

Ask students: What do you think is an engineer? Do you know anyone who is an engineer? What kind of things do they do?

Explain to students that engineers do many different things, one of which is working with and designing computers and robots. In this lesson, they will learn about those kinds of engineers, but first, they need to understand what all engineers have to do: design. Introduce the steps of the Design Process.

An **engineer** is anyone who invents or improves things (for instance, just about any object you see around you) or processes (such as methods) to solve problems or meet needs. Any human-made object you encounter in your daily life was influenced by engineers. There are many different kinds of engineers including: biomedical engineers, aerospace engineers, computer engineers, and industrial engineers.

For descriptions and further activity ideas, check out the following resources:

- http://www.discovere.org/our-programs/engineers-week
- http://www.eie.org/eie-curriculum/curriculum-units
- *Engineering the ABC*'s by Patty O'Brien Novak

Design Process

When making projects, engineers follow a series of steps called the **Design Process**. It has 6 steps: ASK, IMAGINE, PLAN, CREATE, TEST & IMPROVE, and SHARE. The Design Process is a **cycle** – there's no official starting or ending point. You can begin at any step, move back and forth between steps, or repeat the cycle over and over!

Design Process song
(to the tune of "Twinkle, Twinkle")
Ask and imagine, plan and create,
Test and improve and share what we make.
(Repeat)

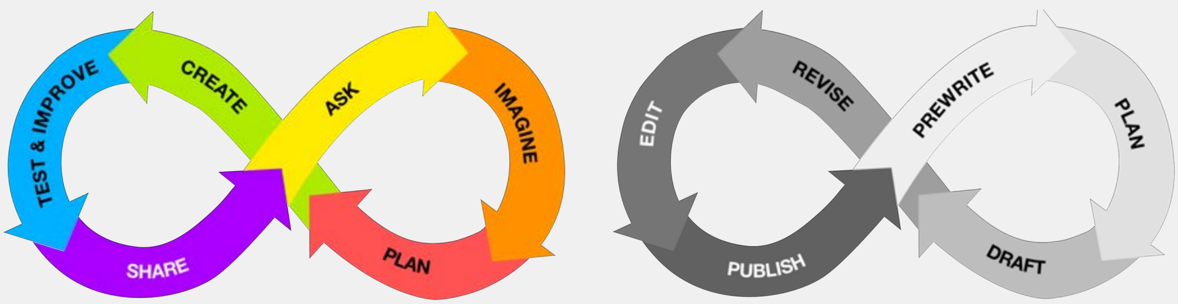
Show students a series of pictures of naturally occurring and man-made objects (show pictures one at a time). Examples of pictures are included in Appendix A. If students think that the object was built by an engineer, they should jump! If they think otherwise, they stay seated. Discuss students' reasoning. *Ask students: What made you think this was built by an engineer? What parts of the object made you think that way?*

ENGINEERS AND WRITERS (10 min)

Show students the Design Process and the Writing Process side by side. Explain to students that both are creative processes that require imagination, planning, creating, revising, feedback, and sharing. Both engineers and writers turn ideas into projects that are shared with others. Ask students what other activities require a process (e.g., cooking, painting, getting good at a sport, etc.). Lead student-centered discussion on the similarities and differences between engineers and writers.

Writing Process

Just as engineers use the Engineering Design Process to design and create projects, writers use the Writing Process to brainstorm ideas, write a draft, make revisions, and share their writing with others. The Writing Process is also a **cycle** – there's no official starting or ending point, and you can move back and forth between steps!



Design Process **Writing Process**

THINK LIKE AN ENGINEER (10 min)

Explain to students that everyone in the class is going to start thinking like an engineer! *Ask students: Have you seen or interacted with robots before? What do they look like? What kinds of different parts make up a robot? How do you think engineers build robots? What might happen if the engineers went straight to building a robot without drawing out a plan first?* The purpose of this activity is to engage students in thinking about design and how engineers use different types of materials to create their products.

HOW TO MAKE A PIZZA (20 min)

How-to-Books are a low-stress entry point into writing. After all, all students know how to do something and the structure of a how-to book is fairly simple. In addition, pictures can easily take the place of words. We even suggest that each step in a how-to book should be accompanied by a sketch or picture.

Pass out the Design Journals. Ask students to create a "How-To Book" for making their own pizza. Ask students to include specific details so that someone else can learn how to make their pizza simply by reading these instructions. Depending on the students' writing level, this activity may need more framing. Ask students: *What topping will you put on top of your pizza? What are the steps for making a pizza?* A wonderful resource for How-To-Books can be found at: https://www.education.com/lesson-plan/creating-a-how-to-book/. Students will share their How-To books in pairs in a later lesson.

Lesson 2: Technological Tools - Robots

Powerful Idea From Computer Science:
Hardware/Software, Representation

Powerful Idea From Literacy:
Tools of Communication

OVERVIEW

The advancement of technology over the years has changed the way people communicate and do things. In this lesson, students will begin to understand how technology and communication tools have evolved. Students share ideas, learn about the different characteristics of robots, and learn about KIBO's programming language.

PURPOSE

By learning to code with the KIBO programming blocks, students understand how programming languages are different from natural spoken languages. Both require clear and precise communication, but while humans can understand many different types of genres of speech, KIBO can only understand commands. Furthermore, understanding that robots have special parts (hardware) to let them follow instructions (software) is a powerful idea of computational thinking, which will help students build more complex programs in subsequent lessons.

ACTIVITIES

- Robot Corners (15 min)
- Characteristics of Robots (10 min)
- Tools of Communication (10 min)
- Human Language vs. Code Language (10 min)
- Programmer Says (15 min)

STUDENTS WILL BE ABLE TO...

- Identify characteristics of a robot
- Compare human languages and programming languages
- Create a simple algorithm using the KIBO programming blocks

PREPARATION FOR TEACHERS

- ☐ Read through the Activity Guide
- ☐ Review the Lesson Slides
- ☐ Go through the KIBO Says cards and take out only the blocks listed in the Materials section
- ☐ Gather all materials for the lesson

MATERIALS

FOR THE TEACHER:

- 1 piece of blank chart paper for the Characteristics of Robots activity*
- 8-10 pictures of robots and non-robots*
- Handwritten and typed message*
- Appendix E for troubleshooting tips

FOR STUDENTS:

- Design Journal (see Appendix C for example)
- KIBO Says cards: Begin and End blocks, blue Motion blocks

*See Lesson Slides and Appendix A for examples

VOCABULARY

- Robot — a machine that can be programmed to do different things
- Barcode — a pattern of lines that are readable by machines (like the KIBO robot)
- Program — a set of instructions for a robot
- Block-based programming language – any programming language that lets users create programs by manipulating blocks versus coded text
- Hardware – the physical parts of a computer or robot (e.g. KIBO body)
- Software – the programs that run on the hardware (e.g., KIBO programming blocks)

Lesson 2: Activity

ROBOT CORNERS (15 min)

As explained in the book *Blocks to Robots* by Dr. Marina Bers (2008, p. 70), **robots** can "refer to a wide range of machines...that take on different forms... and can perform autonomous or preprogrammed tasks". Despite their differences, all robots are "capable of movement under some form of control and can be used to perform physical tasks." For example, you can give the robot a set of instructions for its motors in order to make the robot move. The robotic "brain", just like the human brain, has the programmed instructions that make the robot perform its behaviors. It may be helpful to watch video clips of different types of robots in action such as home robots, space robots, factory robots, hospital robots, and child-made robots.

Ask all students to stand in a line or circle where they can see you. Designate three corners of the classroom: one corner for "Robots", one corner for "Maybe Robots", and one corner for "Not Robots." One at a time, show a variety of different pictures of robots and non-robots (e.g. computers, cars, animals, foods, famous robots such as Wall-E and R2D2). Ask students to move to the corner that they think represents the picture. Then ask a few students to explain why they think the picture is a robot or not a robot or why they think it might be a robot. Do not reveal answers until after the next activity: Characteristics of Robots. It is important in this activity for students to share their ideas about they think a robot is.

CHARACTERISTICS OF ROBOTS (10 min)

Read the true/false statements about robots below. Ask students to stand (or make another movement like snapping or waving their fingers in the air) for statements they think are true and sit down for statements they think are false.

Extended Graphing Activity: As you go along, make a graph on a piece of chart paper with True and False for each question along the horizontal axis and number of students along the vertical axis. Have students place a marker (sticker, symbol, etc.) in the "True" or "False" column. Explain to students that the graph allows us to see whether there were more "True" or "False" responses for each question.

1. *Robots are machines (TRUE).*
2. *All robots are made of the same materials (FALSE).*
3. *Robots must have moving parts (TRUE).*
4. *Robots can think by themselves (FALSE).*
5. *All robots look alike (FALSE).*
6. *Robots must be able to move around the room (FALSE).*
7. *Robots are operated using remote controls (FALSE).*
8. *People tell robots how to behave with a list of instructions called a program (TRUE).*
9. *Some robots can tell what is going on around them (TRUE)*
   (Examples: sensing light, temperature, sound, or a touch.)
10. *Robots are alive (FALSE).*

Choose 1-2 pictures from the Robot Corners activity and lead student-centered discussion about why that picture represents a robot or is not a robot based on what they have just learned about robots.

For further activity ideas on robots, check out the following resources:

- *Robots, Robots Everywhere!* by Sue Fliess
- *National Geographic Readers: Robots* by Melissa Stewart

TOOLS OF COMMUNICATION (10 min)

Have students sit in a circle and play a game of "Telephone", in which one student thinks of a message and whispers it to the person sitting next to them, who then whispers to the person next to them, and so on and so forth until the message gets to the last person. Ask the last person and the first person to say their messages out loud and compare the two messages. *Ask students: Were the two messages the same? Why or why not? What are some other ways we could use to pass along a message?*

Repeat the game, this time by giving each student a typed and printed version of the message. Have a few students read out their printed message. *Ask students: How was this better than the last round? Are all students able to receive the same information?*

At the end of the activity, explain to students how this mirrors the evolution of writing technology from oral societies to scribal writing to post-printing press. Help students draw the connection to the evolution of computers and robotic technologies. More specifically, explain to students that if we had to program robots without writing, it would be messy, but we can use computer writing to program robots, and that is called **code**.

HUMAN LANGUAGE VS. CODE LANGUAGE (10 min)

This activity also has two parts: Meaning of Words and KIBO's Language. Both activities serve to illustrate how human languages (written and spoken) can be used to communicate a variety of things (e.g. sarcasm, allusions, hyperbole/ exaggerations, etc.), whereas programming languages are more structured and literal.

For the Meaning of Words activity, the goal is to remind students of what Mikhail Bakhtin calls, "heteroglosia," the multiple meanings we all carry for each word. In simple terms, human language is much more dynamic than code language. Ask students what people actually mean when they say certain things. For example:

I'm so hungry I could eat a horse!
I have a million things to do today.
My homework is taking forever to get done.

For the KIBO's Language activity, show students the large KIBO Says cards. Have students point out what they see on each block: the text, the icon, colors, the barcode, etc. *Ask students: What part of the block is KIBO's language? Is it the words, or the pictures, or something else?* Once students identify the barcode as the answer, discuss other objects or places where they have encountered barcodes.

Then *ask students: Do you think KIBO can think on its own? Can KIBO make its own program?* Lead student-centered discussion on how robots are programmed by humans and cannot think for themselves. Everything that KIBO says and does is determined by how the programmer chooses the program, or set of instructions, for KIBO. For example, we say we want KIBO to move forward, but KIBO reads the barcodes for the Begin, Forward, and End blocks.

PROGRAMMER SAYS (15 min)

In order to program the KIBO robot, students first need to learn KIBO's language: the programming blocks! This activity is played like the traditional "Simon Says" game, in which students repeat an action if Simon says to do something. Briefly introduce each programming instruction and what it means (use only the blocks listed in the Materials section in this lesson).

What is a Program?

A **program** is a sequence of instructions that the robot acts in order. Each instruction has a specific meaning, and the order of the instructions affects the robot's overall actions. This is an example of a KIBO program.

Have the class stand up. Hold up one big KIBO icon at a time and say, "Programmer says to __________". Go through each individual instruction a few times until the class seems to get it. Once students are familiar with each instruction, ask for volunteers to be the Programmer who gives the class full programs to run through (e.g. Begin, Spin, Forward, End). Just like in the real "Simon Says" game, the Programmer can try to be tricky! For example, if the Programmer forgets to give a Begin or End instruction, should the class still move? Just like Simon Says, if the Programmer forgets to say, "Programmer says to ____________", then students should sit down! This will help reinforce the concept that KIBO is programmed by humans.

Lesson 3: Sequencing

Powerful Idea From Computer Science:
Hardware/Software, Algorithms, Representation

Powerful Idea From Literacy:
Summarizing/Retelling the Sequence of a Story, Descriptive Language in Writing

OVERVIEW

Students will learn about sequencing in programming and think about how it relates to sequencing in literacy, and why order matters in both cases. Once students become familiar with some of the KIBO programming blocks, they will learn about the different parts of the KIBO robot.

PURPOSE

In the previous lesson, students began learning about different KIBO blocks. Now they will engage in goal-oriented programming, in which students purposefully choose actions in a specific order to achieve a particular outcome. Understanding that order matters is an important skill for students not only in computer science and literacy, but also in their everyday lives as they learn to tie their shoelaces, reflect on the day's activities, plan a family vacation, and more.

ACTIVITIES

- Where the Wild Things Are (15 min)
- Writing Reflection (20 min)
- Program the Teacher with KIBO Blocks (10 min)
- Meet the KIBO Robot (10 min)
- Clean up (5 min)

STUDENTS WILL BE ABLE TO...

- Understand why order matters when programming a robot or telling a story
- Identify the different parts of the KIBO robot

PREPARATION FOR TEACHERS

- ☐ Read through the Activity Guide
- ☐ Review the Lesson Slides
- ☐ Go through the KIBO Says cards and take out only the blocks listed in the Materials section
- ☐ Ensure all KIBO bodies have 4 working AA batteries
- ☐ Sort KIBO blocks and pieces (listed in Materials section) by part and place in a central location

MATERIALS

FOR THE TEACHER:

- 1 copy of *Where the Wild Things Are* by Maurice Sendak
- Large KIBO Says cards: Begin and End blocks, blue Motion blocks, Beep and Sing blocks
- 1 flathead screwdriver
- Extra AA batteries
- Appendix E for troubleshooting tips

FOR STUDENTS:

- Design Journal (see Appendix C for example)
- KIBO bodies, wheels, motors, and art platforms

*See Lesson Slides or Appendix A for examples

VOCABULARY

- Algorithm – a list of steps to finish a task
- Instruction – a direction that a robot will understand
- Order – parts of a group arranged in a specific way (e.g., smallest to largest, tallest to shortest)
- Program – a complete set of instructions for a robot
- Scanner – electronic device for reading printed barcodes
- Sequence – to arrange a set of instructions in a particular order
- Main board – the robot's "brain" that has the programmed instructions that the robot to perform its behaviors
- Motor – the part of a robot that makes it move
- Wheels – the round parts of a vehicle that turn in circles and allow it to move

Lesson 3: Activities

WHERE THE WILD THINGS ARE (15 min)

Read the book *Where the Wild Things* Are as a class; if needed, read the book a second time. Lead a student-centered discussion that reviews the events of the story. You can prompt the students: *Who can summarize the main events in this story? (e.g. first he made some mischief, then more, then yelled at his mother, etc.). Then ask students: What if the first scene was Max on a boat? How would that change the story? What about if Max had smelled the food before making more mischief?* The purpose of this activity is to get students to think about sequencing in narrative.

WRITING REFLECTION (20 min)

After you read the story, ask students: *What if the first scene was Max on a boat? How would that change the story?* Have students open up their design journals and write a new beginning to the story that has Max on a boat. The writing prompt is "*Write a new opening to the story with Max on a boat. What does the reader need to know? How did Max get on the boat?*"

Alternatively, you can modify the writing prompt if it is too advanced for your students. After reading the story, have the students use one of the blank Design Journal pages included at the end of the packet to write about one of the following prompts (you should choose one for the entire class):

- The strangest place you ever visited
- A time you were mischievous (define for students) and your parents or teacher got upset with you

Encourage students to write creatively and express their ideas freely in their journals. Feel free to reinforce any writing skills they are learning in other lessons.

PROGRAM THE TEACHER WITH KIBO BLOCKS (10 min)

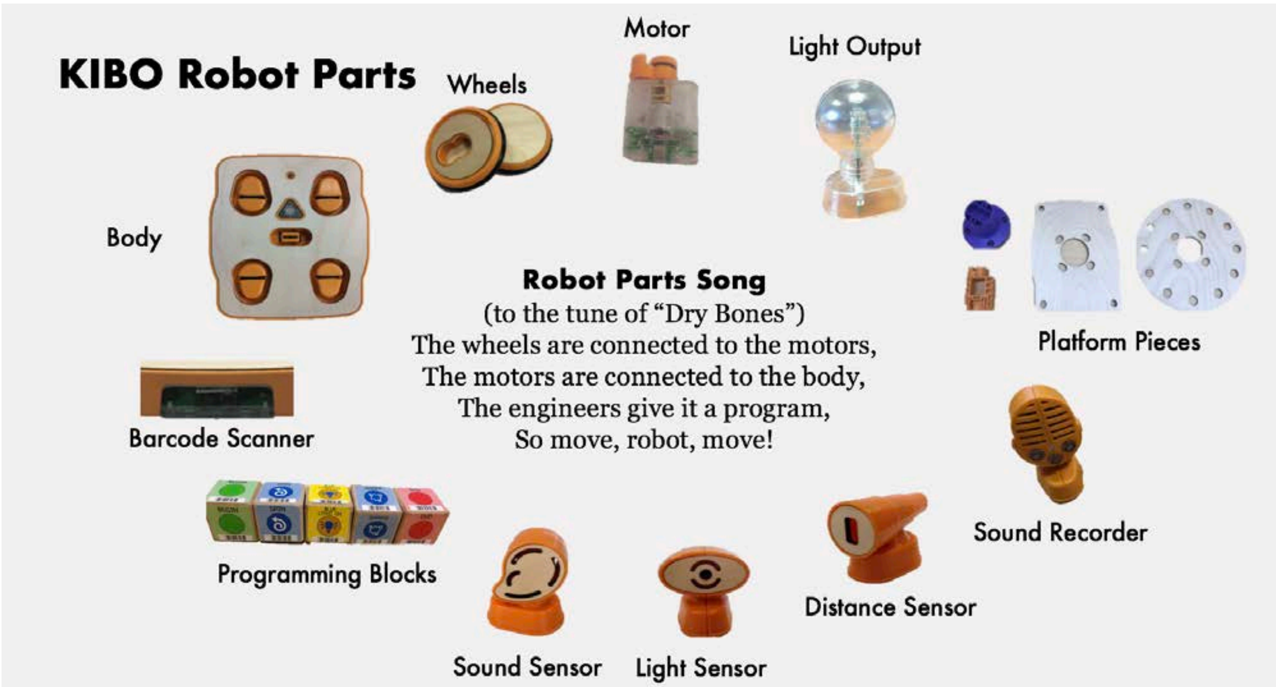
Using the KIBO Says cards, students will work together as a class to "program" their teacher to move from one part of the room to the other. Be silly! An example would be for the students to "program" their teacher to move from the front of the room to the library area by using these blocks: Begin, Forward, Spin, Turn Left, Forward, Forward, End. The goal of this game is for students to practice sequencing as a class before working individually or in their small groups. Before the teacher-robot moves, students can make predictions about where the teacher-robot will end up. It may be helpful to let the students make mistakes in order to foster a discussion on sequencing and debugging.

Remind students: a program is a set of instructions that tells the robot what to do. KIBO's program always starts with the green Begin block and the red End block. Everything you (the programmer) want KIBO to do goes in between those blocks and will happen in the same order that you place the blocks.

MEET THE KIBO ROBOT (10 min)

Take out KIBOs and blocks. Explain to students that today they will be learning how to put together the different parts of the KIBO robot. Show students a KIBO robot body. *Ask students: What parts do you see through the clear backside of KIBO? What do you think those parts do? What do the batteries do? What are some other objects you have seen that have the same function? (e.g. KIBO's wheels are like the tires on a car)*

Using the KIBO parts guide below, introduce the KIBO robot's key parts and their functions. Teach the "Robot Parts Song" and have students sing and dance along. Explain to students that the song helps us understand how to put the KIBO robot together. Demonstrate how to attach the wheels, motors, and art platforms. If time permits, allow students to work in pairs to assemble their own KIBO robot and practice scanning a simple program: Begin, Shake, End.



Refer back to your KIBO guide to help students troubleshoot any issues with KIBO. Prompting questions for students: *How do you turn KIBO on? How do you scan a KIBO block? Does your program have all the blocks you need? How do you know KIBO understood that you scanned the block correctly?* If students are working in pairs, introduce the Job Cards so that students can practice taking turns either scanning the program or assisting the scanner. Examples of Job Cards are included in Appendix A.

CLEAN UP (5 min)

Using your classroom organizational system, have students clean up their work areas and put away all KIBO parts and blocks in their appropriate spots.

Lesson 4: Programming

Powerful Idea From Computer Science:
Algorithms, Design Process

Powerful Idea From Literacy:
Descriptive Language in Writing

OVERVIEW

Students will learn about sequencing in programming and think about how it relates to sequencing in literacy. Students will program KIBO to dance the Hokey Pokey, or if you wish, a different children's song where students can program a robot to dance to the words. At the end of the lesson, students will demonstrate their current level of understanding by completing the first KIBO Mastery Challenge.

PURPOSE

In the previous lesson, students had the opportunity to engage with KIBO's hardware and software separately. Now they will engage in goal-oriented programming, in which students purposefully choose their KIBO blocks and place them in a specific order to achieve a particular outcome.

ACTIVITIES

- Dance the Hokey Pokey (5 min)
- Program the Hokey Pokey (20 min)
- Hokey-Pokey Reflection (10 min)
- Share Creations (5 min)
- Clean Up (5 min)
- KIBO Mastery Challenge A (15 min)

STUDENTS WILL BE ABLE TO...

- Tell and retell a story clearly and effectively
- Identify common errors with scanning KIBO programs and troubleshoot them
- Practice scanning programs with KIBO
- Learn strategies for debugging and editing

PREPARATION FOR TEACHERS

- ☐ Read through the Activity Guide
- ☐ Review the Lesson Slides
- ☐ Print Job Cards*
- ☐ Ensure all KIBO bodies have 4 working AA batteries
- ☐ Sort KIBO blocks and pieces (listed in Materials section) by part and place in a central location
- ☐ Print KIBO Mastery Challenge A (one for each student)

MATERIALS

FOR THE TEACHER:

- Anchor chart of discussion sentence starters*
- Job Cards*
- 1 flathead screwdriver
- Extra AA batteries
- Appendix E for troubleshooting tips

FOR STUDENTS:

- Design Journal (see Appendix C for example)
- KIBO bodies, wheels, motors, Begin and End blocks, blue Motion blocks, yellow Light blocks, Beep and Sing blocks
- KIBO stickers

*See Lesson Slides or Appendix A for examples

VOCABULARY

- Instruction – a direction that a robot will understand
- Program — a complete set of instructions for a robot
- Scanner — electronic device for reading printed barcodes

Lesson 4: Activities

DANCE THE HOKEY POKEY (5 min)

Explain to students that today they will program KIBO to do the Hokey Pokey. Sing and dance the Hokey Pokey as a class to make sure everyone knows and remembers it. Conclude with a "robot verse":

You put your right hand in,
You put your right hand out,
You put your right hand in,
And you shake it all about,

You do the hokey pokey
and you turn yourself around
That what it's all about. (clap, clap!)

2) left hand
3) right foot
4) left foot
5) head
6) whole self

You put your robot in, you put your robot out,
You put your robot in, and you shake it all about.
You do the Hokey Pokey, and you turn yourself around.
And that's what it's all about. (Clap, clap.)

PROGRAM THE HOKEY POKEY (20 min)

Take out KIBOs and blocks. Remind students how to assemble the KIBO blocks and scan a complete program with KIBO. Have several students share out their strategies for scanning KIBO. Individually or in pairs, students program their KIBOs to do the Hokey Pokey. If this activity is the first time students will be working formally in pairs, introduce the Job Cards so that students can practice taking turns either scanning the program or assisting the scanner. Examples of Job Cards are included in Appendix A.

HOKEY POKEY REFLECTION (10 min)

In their Design Journals, ask students to record their Hokey Pokey programs by using the KIBO stickers to write out the blocks in their program. Ask students: *What is your favorite block in your program? What does the block do? How did the block help you program the Hokey Pokey?* Provide an example if needed: *My favorite block is the Spin block. The Spin block makes the robot go around in a circle. I used the Spin block so that the robot turns itself around (because that's what it's all about!)*

SHARE CREATIONS (5 min)

When all groups are done with their Hokey Pokey robot programs, ask the whole class to play their programs at once and dance the Hokey Pokey! This is the first time that students engage in goal-oriented programming. Using the Discussion Sentence Starters anchor chart, ask students about their challenges of programming: *What problems did you have when* *you were scanning blocks? Did you ever get an error message? Did you ever feel frustrated or disappointed? Why did you feel that way?* Note down students' responses on a piece of paper so that you can come back to these points in the next lesson.

CLEAN UP (5 min)

Using your classroom organizational system, have students clean up their work areas and put away all KIBO parts and blocks in their appropriate spots.

KIBO MASTERY CHALLENGE A (15 min)

On the Appendix B-1 you will find challenge A. Please hand out one copy of the assessment to each child in your class.

Instructions:

- Read each question and option out loud to the group. Students can ask to have questions or options read out loud up to 3 times.
- Instruct children to circle only 1 answer per question.
- Make sure students answer the questions by themselves. Students should not be discussing or copying answers.
- Hand in completed answer sheets to Angela de Mik or a member of your school's assessment team.

Lesson 5: Debugging

Powerful Idea From Computer Science:
Debugging

Powerful Idea From Literacy:
Editing, Awareness of Audience

OVERVIEW

In this lesson, students learn the importance of communicating effectively to an audience. Students engage in this learning by retelling a story to their peers and "edit" their story when their audience is confused and needs more clarification. Students connect this idea to when the KIBO robot does not perform the intended instructions. The process of figuring out what went wrong and how to fix things is called debugging.

PURPOSE

The parallel of editing in literacy and debugging in computer science is crucial to students' understanding of the differences between humans and computers/robots. Humans might be able to tell what a storyteller is trying to communicate even if they leave out a few details; however, a computer is far less flexible. Furthermore, this lesson allows students to not only encounter obstacles, but also to identify and troubleshoot these issues, thus building their confidence to tackle later, more challenging lessons.

ACTIVITIES

- Tell a Story (20 min)
- Why is KIBO Confused? (15 min)
- Free Play (10 min)
- Debugging Reflection (10 min)
- Clean Up (5 min)

STUDENTS WILL BE ABLE TO...

- Identify common errors with scanning KIBO programs and troubleshoot them
- Practice scanning programs with KIBO
- Learn strategies for debugging and editing

PREPARATION FOR TEACHERS

- ☐ Read through the Activity Guide
- ☐ Review the Lesson Slides
- ☐ Prepare Why is KIBO Confused? Anchor Chart
- ☐ Ensure all KIBO bodies have 4 working AA batteries
- ☐ Sort KIBO blocks and pieces (listed in Materials section) by part and place in a central location
- ☐ Make copies of students' How-To-Book drafts from Lesson 1

MATERIALS

FOR THE TEACHER:

- Why is KIBO Confused? Anchor Chart
- Job Cards
- 1 flathead screwdriver
- Extra AA batteries
- Appendix E for troubleshooting tips

FOR STUDENTS:

- Design Journal (see Appendix C for example)
- KIBO bodies, wheels, motors, Begin and End blocks, blue Motion blocks, yellow Light blocks, Beep and Sing blocks

*See Lesson Slides or Appendix A for examples

VOCABULARY

- Debug — finding and fixing problems in an algorithm or program
- Edit — to make changes to something
- Persistence – trying again and again, even when something is very hard
- Reboot – to turn off the device and turn it back on
- Troubleshoot – actions taken to solve a problem with hardware or software

Lesson 5 Activities

TELL A STORY (20 min)

Framing for the Teacher: Even the most basic forms of writing (letter and letter-like forms) require a high level of abstraction that speech does not. Education psychologist, Lev Vygotsky explained, “In learning to write, the child must disengage himself from the sensory aspects of speech and replace words by images of words” (Vygotsky, 2012, p.181). Writing requires symbolization of the sound image into written signs (letters, syllables, etc.). It is this abstract quality of written language, specifically double abstraction- abstraction from the sound of speech and abstraction from the interlocutor (the reader)- that makes it challenging. Of course, young children are at the very beginning of this journey. **The goal at this stage of development is to understand that even though all the details exist in your head, if you don’t provide them for your reader, your story won’t make sense. This is the debugging/editing challenge for students at this stage of development.**

Split the class into two groups and have each group work together to write instructions on how to put together KIBO and make it move. Then gather the whole class and have the groups exchange their written instructions. Have each group read the other group’s instructions and explain the task based on what they understood from the instructions. This activity can be fun and light. The big idea is that it can be hard to communicate what’s in our head to someone or something else.

Explain to students that with writing there may be multiple ways to communicate the same thing, and even if we misspell a word or make a grammatical error, our message may still be clear. However, when robots or computers are the audience, we have to make sure to communicate in the way that the machines understand. There is much less margin for error.

WHY IS KIBO CONFUSED? (15 min)

In Lesson 3, students shared challenges of scanning KIBO blocks and other issues that they experienced while creating their Hokey-Pokey programs. Check back on your notes from that discussion and prepare an anchor chart noting 4-5 of these challenges on the left side of the chart, leaving the right side empty for students to provide solutions in this activity.

Present the anchor chart to students. Explain to students how in the previous lesson, students encountered different challenges, such as not being able to scan the blocks properly, seeing a red light or hearing a minor key sound when scanning the blocks, etc. Other examples of common errors can be found in this KIBO troubleshooting tip sheet: http://kinderlabrobotics.com/wp-content/uploads/2017/10/KIBO-10-Quick-Start-Guide.pdf.

Ask students to brainstorm 1-2 solutions for every solution. An example is provided below:

Challenge #1: It’s hard to scan the blocks.	Solution #1: Separate the blocks instead of connecting the pegs. Scan each block individually. Solution #2: Ask your partner to cover the other barcodes on the left and right of the block you’re trying to scan.
Challenge #2: When I accidentally scan the End block twice, it gives me a red light, and I have to scan the program all over again.	Solution #1: Tilt the KIBO immediately after scanning the block so that the barcode scanner doesn’t accidentally scan it twice.

Explain to students that **debugging** is a method used to understand how to fix things when engineers program robots, and the robots do not work. By identifying these problems and different solutions to solve them, students are debugging.

Debugging is a word used in computer science to describe when people find errors in their computer programs and use different strategies to solve the problem. While the word "bug" was used in other scientific fields, the word "debugging" is attributed to Admiral Grace Hopper, who back in the 1940s found a moth stuck inside the computer (computers used to be that big!), which caused an error in the system. She was able to resolve the error by taking out the bug, hence the word "debugging"!

For further activity ideas and examples of pictures, check out the following resources:

- https://www.computerhope.com/issues/ch000984.htm
- https://www.npr.org/sections/alltechconsidered/2015/11/23/457129179/the-future-of-nanotechnology-and-computers-so-small-you-can-swallow-them

FREE PLAY (10 min)

Take out KIBOs and blocks. This activity is a great opportunity for students to freely explore with the KIBO robot and the programming blocks. Encourage students to try to make these mistakes purposefully and to practice debugging! By the end of this activity, students should feel comfortable scanning a complete program with KIBO.

DEBUGGING REFLECTION (10 min)

Pass out students' Design Journals. Ask students to reflect about one of the problems they had with KIBO. *What was the problem?* Ask students to explain why KIBO wasn't understanding what they wanted KIBO to do. *How did you change the way you scanned (communicated) so that KIBO would understand?* Students can reflect in their Design Journals by drawing a picture of how they debugged, or if they can, write about their problem solving strategy.

CLEAN UP (5 min)

Using your classroom organizational system, have students clean up their work areas and put away all KIBO parts and blocks in their appropriate spots.

Lesson 6: Cause and Effect - Level 1

Powerful Idea From Computer Science:
Control Structures, Representation, Sensors

Powerful Idea From Literacy:
Spelling-Sound Correspondence

OVERVIEW

In this lesson, students will learn about cause and effect and sensors by being introduced to two new modules: the Sound Sensor and the Sound Recorder. The Sound Sensor uses an event (wait for clap) before performing the subsequent action. The Sound Recorder allows students to record up to three different sound clips, depending on the shape of the recorder button they press (circle, square, triangle). Students will learn that for KIBO to play their sound clip, they must use the orange Sound Recorder block that has the same shape (circle, square, triangle).

PURPOSE

When students learned to program with the Beep and Sound blocks in previous lessons, KIBO did not require a sound sensor because those sounds were produced from the robot (output). In this lesson, students learn how robots can take in information from the environment to then perform an action (input). These concepts are integral to the understanding of control structures, which will prove useful in subsequent lessons.

ACTIVITIES

- What did Max Sense (15 min)
- KIBO Sound Sensor (5 min)
- Shape Shifting (10 min)
- KIBO Sound Recorder (5 min)
- Free Play (10 min)
- Clean Up (5 min)
- KIBO Mastery Challenge B (10 min)

STUDENTS WILL BE ABLE TO...

- Distinguish between human senses and robot sensors
- Use the KIBO Sound Sensor with its appropriate Wait for Clap block
- Record a sound clip successfully using the Sound Recorder module and Sound Recorder blocks

PREPARATION FOR TEACHERS

- ☐ Read through the Activity Guide
- ☐ Review the Lesson Slides
- ☐ Ensure all KIBO bodies have 4 working AA batteries
- ☐ Sort KIBO blocks and pieces (listed in Materials section) by part and place in a central location
- ☐ Print KIBO Mastery Challenge B (one for each student)

MATERIALS

FOR THE TEACHER:

- 1 copy of *Where the Wild Things Are* by Maurice Sendak
- KIBO Says cards: orange Sound Recorder blocks
- Job Cards
- 1 flathead screwdriver
- Extra AA batteries
- Appendix E for troubleshooting tips

FOR STUDENTS:

- Design Journal (see Appendix C for example)
- Construction paper and markers
- KIBO bodies, wheels, motors, Begin and End blocks, blue Motion blocks, yellow Light blocks, Beep and Sing blocks, Wait for Clap blocks, Sound sensors, Sound Recorder module, orange Sound Recorder blocks

VOCABULARY

- Attribute – the quality or characteristic of something (e.g. an attribute of sound blocks is the color orange)
- Senses — the way humans and animals take in information about the surrounding environment. Humans have five senses: touch, taste, smell, sight, and hearing
- Sensor — a special part that helps machines take in information about the surrounding environment; there are sensors that are very much like human senses
- Event — an action that causes something to happen
- Circle — a round, closed shape with no edges
- Record — to make something (like a sound) permanent so that it can be played back at a later time
- Sound — a type of energy made by vibrations in the air that we can hear
- Square — a closed shape with four equal sides
- Triangle — a closed shape with three sides

WHAT DID MAX SENSE? (15 min)

Throughout the *Where the Wild Things Are* story, Max uses his five senses: taste, smell, touch, hearing, and sight. *Ask students: What body parts do humans use to sense things in our environment?* As a class, decide on a movement that represents each of the five senses. For example, you might decide to point to your tongue for taste, nose for smell, fingers for touch, ear for hearing, and eyes for sight.

At your discretion, reread the story as a class, or select pages from the story. As you read, pause at different points and ask students to do the movement that corresponds to what Max is sensing. This is an important chance to point out to students that literature "shows" instead of "tells." For example, when we read, "The wild things roared their terrible roars and gnashed their terrible teeth and rolled their terrible eyes and showed their terrible claws," we understand what exactly Max was seeing (and maybe even smelling if they had bad breath with those terrible teeth!). Explain to students how literature sometimes uses "poetic" language, whereas computer science uses literal and command-oriented code language.

Below are examples of quotes from the story to pause and have students identify Max's senses:

Taste	*"So he was sent to bed **without eating anything**."*
Smell	*"Then all around from far away, across the world, he **smelled good things to eat**, so he gave up being King of where all the Wild Things are"*
Touch	*"The night Max wore his **wolf suit** and made mischief of one kind and another"* *"and into the night of his very own room where he found his supper waiting for him and **it was still hot**."*
Hearing	*"His mother **called him 'WILD THING'**"* *"And they were frightened and **called him the most Wild Thing of all** and made him King of all Wild Things"* *"But the **wild things cried**, "Oh please don't go we'll eat you up-we love you so!" And **Max said**, "No!"* *"The wild things **roared their terrible roars** and gnashed their terrible teeth and rolled their terrible eyes and showed their terrible claws, but Max stepped into his private boat and waved goodbye"*
Sight	*"That very night in Max's room, **a forest grew**..."* *"When he came to the place where the Wild Things are...**terrible claws**"* *"Till Max said BE STILL...**blinking once**"* *"The wild things roared their terrible roars and **gnashed their terrible teeth and rolled their terrible eyes and showed their terrible claws**, but Max stepped into his private boat and **waved goodbye**"* *"and into the night of his very own room where he **found his supper waiting for him** and it was still hot."*

KIBO SOUND SENSOR (5 min)

Take out KIBOs and blocks. Show the Wait for Clap block and the Sound sensor and create an example program together. Run the program, and have students discuss what the robot is doing. Introduce the term event, which is an action that causes something to happen. The action here is the clap, which causes KIBO to continue its program. All of the sensors that KIBO has (sound, light, and distance) use events to trigger KIBO which they will experiment with in later lessons.

What is the Sound Sensor?

KIBO's **Sound Sensor** is shaped like an ear and senses sounds from the environment. It is programmed using the Wait for Clap block. In the example program, KIBO will turn right, wait for a loud sound (like a clap) before it spins and ends.

SHAPE SHIFTING (10 min)

Show students the large KIBO Says cards depicting the three orange Sound Recorder blocks: Play Circle, Play Square, and Play Triangle. First demonstrate the game as a class. Ask for three student volunteers to choose a sound or action to go with each shape. For example, a student might decide to jump on one foot for the Circle, another student might decide to yell "Hooray!" for the Triangle, and a third student might decide to hold up a book for the Square. To play the game, explain to students that when you hold up one of the three shape cards, they should only perform the action associated with that shape.

Once students are comfortable with the game, split into small groups and have students take turns deciding the actions for the shapes. Have students recreate the shape cards using construction paper and markers. The purpose of this activity to get students comfortable with cause and effect; students must shift or alter their actions depending on the shape of the card. This will also help students better understand how the KIBO Sound Recorder module works in the next demonstration activity.

KIBO SOUND RECORDER (5 min)

Show students the KIBO Sound Recorder module. Demonstrate with a model program how they can make three different recordings by pressing and holding down on the three shape buttons on the module. Note that the Sound Recorder must be connected to power by inserting the module into the KIBO body before recording.

What is the Sound Recorder?

KIBO's sound recorder/playback module has three different buttons – square, triangle, and circle– that allows students to record three different short sound clips. Remember to match the shape of the block to the recorded sound!

FREE PLAY (10 min)

Individually or in pairs, students should take this time to explore the Sound Sensor and Sound Recorder modules freely. By the end of this free-exploration, students should understanding the difference between sound input (i.e., KIBO needs to hear the clap using the Sound Sensor before proceeding) and sound output (i.e., students record the specific sounds that they want KIBO to play using the Sound Recorder). Encourage students to try other noises, like stomping or ringing a bell, to trigger the Sound Sensor!

CLEAN UP (5 min)

Using your classroom organizational system, have students clean up their work areas and put away all KIBO parts and blocks in their appropriate spots.

KIBO MASTERY CHALLENGE B (10 min)

On the Appendix B-2 you will find challenge B. Please hand out one copy of the assessment to each child in your class.

Instructions:

- Read each question and option out loud to the group. Students can ask to have questions or options read out loud up to 3 times.
- Instruct children to circle only 1 answer per question.
- Make sure students answer the questions by themselves. Students should not be discussing or copying answers.
- Hand in completed answer sheets to Angela de Mik or a member of your school's assessment team.

Lesson 7: Cause and Effect - Level 2

Powerful Idea From Computer Science:
Algorithms, Modularity, Representation

Powerful Idea From Literacy:
Descriptive Language in Writing

OVERVIEW

Students will engage in goal-oriented programming using the Sound Sensor and Sound Recorder modules. In this lesson, students combine their programming knowledge from previous lessons to program their robots to sing and dance to the "If You're Happy and You Know It" song.

PURPOSE

This lesson reinforces students' learning of the KIBO blocks and the use of different KIBO modules. At this point in the curriculum, students should be familiar with the Lightbulb, Sound Sensor, and Sound Recorder modules. The focus of the project reflection time will shift from discussing debugging issues (though that, of course, leads to thoughtful discussion) to goal-oriented programming and how students' initial ideas and plans might not always translate to their final KIBO program.

ACTIVITIES

- Sing "If You're Wild and You Know It" (5 min)
- Program "If You're Wild and You Know It" (30 min)
- Project Reflection (10 min)
- Share Creations (10 min)
- Clean Up (5 min)

STUDENTS WILL BE ABLE TO...

- Program KIBO to sing and dance to the "If You're Wild and You Know It" song

PREPARATION FOR TEACHERS

- ☐ Read through the Activity Guide
- ☐ Review the Lesson Slides
- ☐ Ensure all KIBO bodies have 4 working AA batteries
- ☐ Sort KIBO blocks and pieces (listed in Materials section) by part and place in a central location

MATERIALS

FOR THE TEACHER:

- Job Cards
- 1 flathead screwdriver
- Extra AA batteries
- Discussion Sentence Starters anchor chart*
- Appendix E for troubleshooting tips

FOR STUDENTS:

- Design Journal (see Appendix C for example)
- KIBO bodies, wheels, motors, Begin and End blocks, blue Motion blocks, yellow Light blocks, Beep and Sing blocks, Wait for Clap blocks, Sound Sensors, Sound Recorder module, orange Sound Recorder blocks
- KIBO stickers

*See Lesson Slides or Appendix A for examples

VOCABULARY

- Senses — the way humans and animals take in information about the surrounding environment. Humans have five senses: touch, taste, smell, sight, and hearing
- Sensor — a special part that helps machines take in information about the surrounding environment; there are sensors that are very much like human senses
- Event — an action that causes something to happen
- Circle – a round, closed shape with no edges
- Record — to make something (like a sound) permanent so that it can be played back at a later time
- Sound — a type of energy made by vibrations in the air that we can hear
- Square — a closed shape with four equal sides
- Triangle — a closed shape with three sides

Lesson 7: Activities

IF YOU'RE WILD AND YOU KNOW IT (5 min)

In the story *Where the Wild Things Are*, Max acts wild. Students will work individually or in pairs to program KIBO to dance a version of the song "If You're Happy and You Know It," Wild-fied, "If You're Wild and You Know It". Sing the song together as a class.

If you're wild and you know it, clap your hands (clap-clap)
If you're wild and you know it, clap your hands (clap-clap)
If you're wild and you know it, then your face will surely show it
If you're wild and you know it, clap your hands. (clap-clap)

2. stomp your feet
3. shout "Hooray!"
4. do all three (clap-clap, stomp-stomp, "Hoo-ray!")

PROGRAM "IF YOU'RE WILD AND YOU KNOW IT" (30 min)

Take out KIBOs and blocks. Before students begin programming KIBO to dance, have students take out their Design Journals and write a response to the following prompt: *What do you wish your robot would do to show that she/he is happy (e.g., dance a particular dance, sing a particular song)*? Have a few students read out their responses and then explain that today KIBO will follow the instructions of the classic song "If You're Happy and You Know It," but in a few lessons, the students will get to decide for themselves what KIBO does to show she/he's happy!

Students should program their robots to move in any wild way during the lyrics "If You're Wild and You Know It" but include the program instructions that have KIBO wait to hear a clap (representing the lyrics "Clap your Hands") before KIBO begins moving. Students can choose as few or as many blocks as they would like to put after the "Wait for Clap" block.

This is an example of a program that teachers can use as a model:

By this point, students have had several opportunities to work in pairs or small groups. Feel free to continue using the Job Cards to enable collaboration and ensure that each student has the opportunity to be hands-on with KIBO.

PROJECT REFLECTION (10 min)

Before sharing their projects, have students take out their Design Journals and use the KIBO stickers to write out their program. Where did they choose to place the Wait for Clap block? What was fun or challenging about creating their program? Did their program get KIBO to do what they wanted? Students should document their reflections in their Design Journals and are encouraged to bring their reflections to share in the Technology Circle.

SHARE CREATIONS (10 min)

Have students sit in a technology circle to share their programs. Encourage students to verbalize their thinking and reasoning behind their program. For example, *ask students: Where did you decide to add the Wait for Clap block? What were the different ways you tried to trigger the Sound sensor (clapping, talking, etc.)? What kinds of sounds did you record? Why did you choose a particular block in your program?* Students can also use the discussion sentence starters from the anchor chart to talk about their creations.

CLEAN UP (5 min)

Using your classroom organizational system, have students clean up their work areas and put away all KIBO parts and blocks in their appropriate spots.

Lesson 8: Repeat Loops - Level 1

Powerful Idea From Computer Science:
Control Structure, Modularity

Powerful Idea From Literacy:
Repetition as a Literary Device, Repetition in Word Forms

OVERVIEW

In this lesson, students understand the importance of repetition both in computer science and literature. Students will learn about a new instruction that makes KIBO repeat programming instructions infinitely or a given number of times. Students also think about repetition as a literary device and the purpose it serves in a text, as well as repetition in word structure as a review of foundational phonic and word recognition skills.

PURPOSE

The activities in this lesson broaden students' understanding of patterns by highlighting the different ways that repetition can be used to make something more efficient or more entertaining. Students also begin to learn that there are multiple ways of representing the same outcome, and that repeat loops are one way that computer scientists make more efficient programs.

ACTIVITIES

- Repetition in Stories and Songs (15 min)
- Toothbrush Exercise (15 min)
- KIBO Repeat with Numbers (15 min)
- Clean Up (5 min)
- KIBO Mastery Challenge C (10 min)

STUDENTS WILL BE ABLE TO...

- Identify patterns in code sequences and rewrite codes using repeat loops
- Use KIBO number parameters to make a program that loops a certain of times
- Understand how repetition is used in stories and songs

PREPARATION FOR TEACHERS

- ☐ Read through the Activity Guide
- ☐ Review the Lesson Slides
- ☐ Ensure all KIBO bodies have 4 working AA batteries
- ☐ Sort KIBO blocks and pieces (listed in Materials section) by part and place in a central location
- ☐ Print KIBO Mastery Challenge C (one for each student)
- ☐ Print Toothbrush Sequence Pictures*

MATERIALS

FOR THE TEACHER:

- KIBO Says cards
- Job Cards
- Toothbrush Sequence Pictures*
- 1 flathead screwdriver
- Extra AA batteries
- Appendix E for troubleshooting tips

FOR STUDENTS:

- Design Journal (see Appendix C for example)
- Small KIBO block cutouts (one set for each pair of students)
- KIBO bodies, wheels, motors, Begin and End blocks, blue Motion blocks, yellow Light blocks, Beep and Sing blocks, Wait for Clap blocks, Sound sensors, Sound Recorder module, orange Sound Recorder blocks, Repeat and End Repeat blocks, number parameters

*See Lesson Slides or Appendix A for examples

VOCABULARY

- Loop – the action of doing something over and over again
- Parameter – a value or limit given to a robot that can be changed (e.g., programmer sets the limit for how many times a robot repeats a sequence)
- Pattern – a design or sequence that repeats
- Repeat – to do something again

Lesson 8: Activities

REPETITION IN STORIES AND SONGS (15 min)

Throughout the book *Where the Wild Things Are*, certain phrases and words are repeated multiple times. Reread the "terrible pages" and come up with a list of what phrases or words are repeated (e.g., terrible, wild things, eat you up, etc.). *Ask students: Why might the author have done that? What purpose does that serve for the reader?* The purpose of this activity is to remind students that repetition is essential in language, literature, and, as they will learn today, coding as well.

OR

If you feel the students/class could use a break from *Where the Wild Things Are*, choose a song the students like instead. Hand out the lyrics to the class, play the song for the class, and ask students, as the song is playing, to circle repeating stanzas. The purpose of this activity is to remind students that repetition is essential in language, literature, and, as they will learn today, coding as well.

TOOTHBRUSH EXERCISE (15 min)

Have students think about the way they brush their teeth. *Ask students: Are there actions that you have to repeat? (e.g. moving the toothbrush from left to right) Are there motions that only happen once? (e.g. squeezing out toothpaste)* Hand out Toothbrush Sequence Pictures to children. Working in pairs, have students arrange the pictures in order, noting which pictures should be repeated. Then have students tape or glue the pictures in order in their Design Journals. Once pairs finish, ask students: *Are there steps missing? What are the missing steps and where do they fit in the sequence? Draw an arrow and describe the step that is missing.* If needed, provide an example of how to draw an arrow between two pictures to write down the missing step.

KIBO REPEAT WITH NUMBERS (15 min)

Take out the KIBOs and blocks. Using the large KIBO Says cards first, show students a sample KIBO program that has repeating blocks (see examples below). *Ask students: What is the repeating pattern in this program? How many times does it repeat?*

Identifying Patterns

In the program below, the repeating pattern is [White Light, Beep] and occurs four times.

In the program below, the repeating pattern is [Spin, Wait for Clap, Sing] and occurs twice. Note that the White Light block is not part of the repeating pattern.

As a class, look back at your example KIBO programs with repeating patterns. *Ask students: Is there a way I could make this program shorter?* Demonstrate to students that the Repeat and End Repeat blocks can be used to make programs that are shorter and more efficient.

Make a sample program using the Repeat blocks and the Repeat Forever parameter card. Emphasize that the robot only repeats the instructions in between the Repeat and the End Repeat blocks. Note to students how the robot will not stop unless you press the button (to stop it). Try another model program using the Repeat 2, 3, or 4 parameters.

Distinguish this kind of repetition from literature, where a repetition may take place pages apart and can include slight variations. For example, in the story *Where the Wild Things Are*, the word "terrible" was repeated several times in different parts of the story.

What is a Repeat Loop?

Repeat and End Repeat are like the bread of a sandwich. The programming blocks put inside of them are like the filling. KIBO will only repeat commands that are placed inside of the **Repeat Loop** sandwich. Segments of the code placed outside of the sandwich will not be repeated.

Parameters are used to tell the robot how many times to repeat, or when to stop repeating. In the program below, KIBO will shake 4 times.

Have students explore their own programs using the Repeat blocks. The emphasis here should be using proper syntax, rather than scanning the program onto KIBO. One suggestion for this activity is to have students create their KIBO programs using the blocks first. Then, students can move to a testing station in a designated location in the classroom, where they can test to make sure their programs work.

CLEAN UP (5 min)

Using your classroom organizational system, have students clean up their work areas and put away all KIBO parts and blocks in their appropriate spots.

KIBO MASTERY CHALLENGE C (10 min)

On the Appendix B-3 you will find challenge C. Please hand out one copy of the assessment to each child in your class.
Instructions:

- Read each question and option out loud to the group. Students can ask to have questions or options read out loud up to 3 times.
- Instruct children to circle only 1 answer per question.
- Make sure students answer the questions by themselves. Students should not be discussing or copying answers.
- Hand in completed answer sheets to Angela de Mik or a member of your school's assessment team.

Lesson 9: Repeat Loops - Level 2

Powerful Idea From Computer Science:
Control Structures, Debugging, Sensors

Powerful Idea From Literacy:
Descriptive Language, Perspectives in Narrative

OVERVIEW

In this lesson, students will learn about the Light and Distance sensors and build upon their understanding of Repeat Loops. Students will program their robots to perform different actions using the Repeat blocks with number and sensor parameters. Based on students' level of understanding, educators should feel free to introduce only one of the sensors. Light and Distance sensors can be tricky, as it can be challenging to create the appropriate environment to trigger the sensors.

PURPOSE

Robots use sensors to gather information from their environment. Students will experiment with the Light and Distance sensors and continue to reflect upon human senses and how they differ from robot sensors. Free exploration of the sensors will allow students to test the sensitivity of the sensors by setting up different types of environments.

ACTIVITIES

- My Five Senses (20 min)
- KIBO Repeat with Light Sensor (10 min)
- Free Play with Repeats (25 min)
- Clean Up (5 min)

STUDENTS WILL BE ABLE TO...

- Compare and contrast human senses and robot sensors
- Successfully test a KIBO program using the Light and Distance sensors

PREPARATION FOR TEACHERS

- ☐ Read through the Activity Guide
- ☐ Review the Lesson Slides
- ☐ Print Design Journals (one for each student)*
- ☐ Ensure all KIBO bodies have 4 working AA batteries
- ☐ Sort KIBO blocks and pieces (listed in Materials section) by part and place in a central location

MATERIALS

FOR THE TEACHER:

- 1 copy of *Where the Wild Things Are* by Maurice Sendak
- Job Cards
- 1 flathead screwdriver
- Extra AA batteries
- Appendix E for troubleshooting tips

FOR STUDENTS:

- Design Journal (see Appendix C for example)
- KIBO bodies, wheels, motors, Begin and End blocks, blue Motion blocks, yellow Light blocks, Beep and Sing blocks, Wait for Clap blocks, Sound sensors, Sound Recorder module, orange Sound Recorder blocks, Repeat and End Repeat blocks, number parameters, Light and Distance sensors, gray sensor parameters

*See Lesson Slides or Appendix A for examples

VOCABULARY

- Distance - the amount of space between two things or people
- Senses – the way humans and animals take in information about the surrounding environment. Humans have five senses: touch, taste, smell, sight, and hearing
- Sensor – a special part that helps machines take in information about the surrounding environment; there are sensors that are very much like human senses
- Event – an action that causes something to happen

Lesson 9: Activities

MY FIVE SENSES (20 min)

Read the story *My Five Senses* by Aliki and discuss scenarios in which students might use each of their five senses. Have students talk in pairs or have a whole-group discussion on different situations in which students would need to use each of the five senses.

KIBO REPEAT WITH LIGHT SENSOR (10 min)

Take out the KIBOs and blocks. Explain to students that KIBO has some of the same senses that we (the ability to "see" light and dark, to feel touch, to speak) and that today they will learn how KIBO is able to "see" light and dark). Explain that KIBO needs special programming instructions to tell KIBO what to do with the information from its Light sensor. Show the Repeat and End Repeat blocks, which are now familiar, and the new Until Light/Until Dark parameter cards. Create two example programs together, one which uses the Until Light parameter and the other with the Until Dark parameter. Run the programs, and have students discuss what the robot is doing in each scenario.

What is the Light Sensor?

KIBO's **Light Sensor** can detect light in the room around it. If a flashlight is shining on KIBO, the light sensor will tell KIBO that it is bright. If there are no lights shining on KIBO, the light sensor will tell KIBO that it is dark.

FREE PLAY WITH REPEATS (25 min)

Individually or in pairs, students will create programs using the Light sensor. Free play with the sensor will allow students to tinker and explore the sensitivity of the sensor. Students can shine a flashlight to trigger the Light sensor. Emphasize that the Repeat blocks with sensor parameters mean that KIBO will continue to perform the actions inside of the Repeat loop until the environment changes to the specific parameter.

CLEAN UP (5 min)

Using your classroom organizational system, have students clean up their work areas and put away all KIBO parts and blocks in their appropriate spots.

Lesson 10: Repeat and If Statements

Powerful Idea From Computer Science:	**Powerful Idea From Literacy:**
Control Structures, Debugging, Sensors	Identify Conflict and Resolution, Making Predictions

OVERVIEW

In this lesson, students build upon their explorations of the Light and Distance sensors but this time with conditional statements, in which the sensor checks for the state of the environment only once before the robot decides which action to perform. Students will program their robots to take different actions based on the state of the sensor.

PURPOSE

This lesson allows students to connect their understanding of branched programs with the key literacy concepts of cause and effect and making predictions. Knowing that outcomes can vary depending on the circumstances is an important concept in early childhood, as students begin to comprehend how decisions are made in everyday life.

ACTIVITIES

- Write an Alternative Story (20 min)
- KIBO Repeat With Distance Sensor (15 min)
- Free Play (20 min)
- Optional Extension Activity: KIBO If Statements
- Optional Extension Activity: KIBO Nested Statements
- Clean Up (5 min)

STUDENTS WILL BE ABLE TO...

- Successfully test a conditional KIBO program using the Distance and Light sensors
- Identify situations that would require an If statement or a Repeat loop

PREPARATION FOR TEACHERS

- ☐ Read through the Activity Guide
- ☐ Review the Lesson Slides
- ☐ Ensure all KIBO bodies have 4 working AA batteries
- ☐ Sort KIBO blocks and pieces (listed in Materials section) by part and place in a central location

MATERIALS

FOR THE TEACHER:

- 1 copy of *Where the Wild Things Are* by Maurice Sendak
- Job Cards
- 1 flathead screwdriver
- Extra AA batteries
- Appendix E for troubleshooting tips

FOR STUDENTS:

- Design Journal (see Appendix C for example)
- Flashlights (one per pair of students)
- KIBO bodies, wheels, motors, Begin and End blocks, blue Motion blocks, yellow Light blocks, Beep and Sing blocks, Wait for Clap blocks, Sound sensors, Sound Recorder module, orange Sound Recorder blocks, Repeat and End Repeat blocks, number parameters, Light and Distance sensors, gray sensor parameters, If and End If blocks, purple sensor parameters

VOCABULARY

- Branched program — a program with two or more possible sequences; the computer/robot makes its decision based on an event
- Conditional — only happens sometimes
- Event — an action that causes something to happen

Lesson 10: Activities

WRITING AN ALTERNATIVE STORY (20 min)

The purpose of this activity to have students think creatively about what could have happened in *Where the Wild Things Are* if Max had done things differently. If necessary, reread the story to students.

Below are some examples from the story. Ask students to think about these hypothetical scenarios, and have several students share out their hypotheses.

- *What would have happened if Max hadn't felt wild and yelled?*
- *If Max had made mischief of one kind, but not another...*
- *If Max had not responded to his mother, "I'll eat you up,"...*
- *If Max had been afraid of the wild things, then...*
- *If Max had never stopped the wild rumpus, then...*

Now the students have the opportunity to turn these suggested alternative stories into compositional texts.

Students will write their alternative stories in their Design Journals. This activity is also an opportunity to review whatever skills the students have most recently learned in writing (e.g. strategies for organization, capitalization of proper nouns, etc.).

KIBO REPEAT WITH DISTANCE SENSOR (15 min)

Take out the KIBOs and blocks, then introduce the Distance sensor. Create two example programs together, one which uses the Until Near parameter and the other which uses the Until Far parameter. Run the programs, and have students discuss what the robot is doing.

What is the Distance Sensor?

KIBO uses the **Distance Sensor** to see how near or far KIBO is from other objects. With Distance Parameters, the Distance Sensor can be used with Repeat Loops to control how KIBO moves.

FREE PLAY (20 min)

Individually or in pairs, students will create programs using the Distance sensor, just as they had in the previous lesson, with the Light sensor. Free play with the sensor will allow students to tinker and explore the sensitivity of the sensor. Students can place objects in front of the robot, triggering the Distance sensor. Emphasize that the Repeat blocks with sensor parameters mean that KIBO will continue to perform the actions inside of the Repeat loop until the environment changes to the specific parameter.

OPTIONAL EXTENSION ACTIVITY: KIBO IF STATEMENTS

If students need more time exploring repeats with sensors, feel free to skip this activity. This is an extension activity for students who have mastered repeat loops with sensors and are ready to learn conditionals with the Begin If and End If blocks.

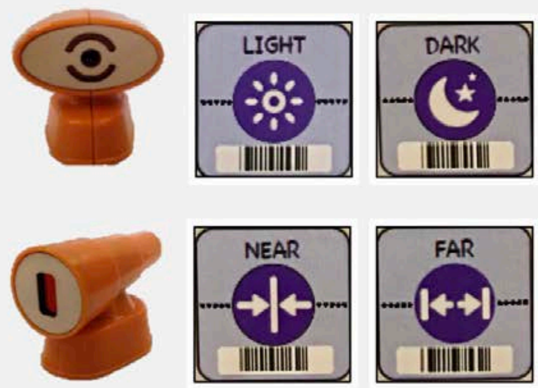
Explain to students that in the programs they have learned so far, KIBO has only one choice of what instructions to do next. Now they will learn an instruction that gives KIBO two choices, and the Light and Distance sensors will help KIBO decide which set of instructions to follow each time the program is run.

Introduce the If and End If blocks, as well as the Near/Far and Light/Dark parameters. Demonstrate what happens when you do and do not put your hand in front of the Distance sensor when the Near/Far parameters are used. Create another program using the Light sensor and Light/Dark parameters and demonstrate what happens when you do and do not shine a light into the sensor.

What is an If Statement?

If Statements allow KIBO to make a choice based on what it can sense, just like your students can! Use these four parameters with If Statements. Remember to attach the appropriate sensors!

Note that unlike Repeat Loops where KIBO will keep repeating the commands inside the loop until it senses something, KIBO's sensor will only check one time before moving to the next part of the program.



Discuss situations in the real world where someone may have to make a choice depending on the circumstances of the environment. For example, "If it is rainy out, I will bring an umbrella". Connect this idea to the Writing an Alternative Story activity, in which students came up with creative endings to the *Where the Wild Things Are* story.

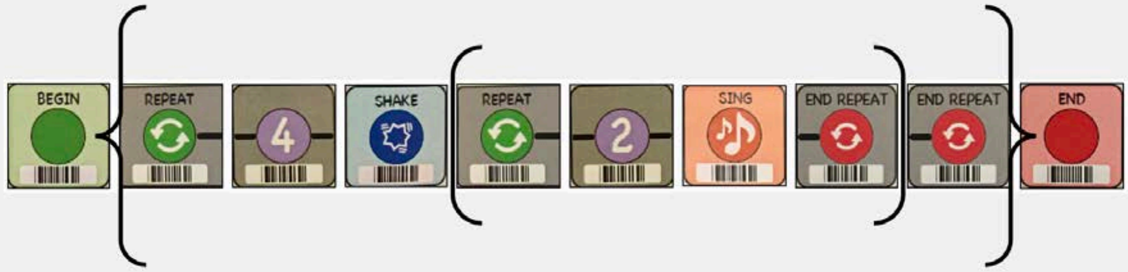
OPTIONAL EXTENSION ACTIVITY: NESTED STATEMENTS

This is an extension activity for students who have mastered single repeat loops and conditional statements. If time permits and students are ready for more advanced learning with KIBO, then show examples of nested loops and nested statements. If students need more time exploring repeats and ifs separately, feel free to skip this activity.

These are the most complex kinds of programs students can create using KIBO. **Nested loops** are a way to make even more efficient programs, in which parts of instructions are repeated a different number of times. Show an example of a nested loop to students. *Ask students: Which blocks are in the* ***inner*** *loop? How many times does it repeat? Which blocks are in the* ***outer*** *loop? How many times does it repeat?*

What is a Nested Loop?

A **nested loop** is a loop inside of another loop. For instance, in the program below, KIBO will shake and beep twice, and repeat this sequence four times.



Just like there can be a repeat inside of another repeat, nested statements are created using any kind of statement (if or repeat) inside of another statement (if or repeat). Before showing the example of a nested statement with KIBO, have students imagine how an automatic faucet works. *Ask students: What happens when you put your hand close to the sensor? Does this happen every time you make that movement?* Walk through the scenario with students: Every time (repeat forever) you move your hands close to the sensor (if), the water runs (end if) (end repeat).

Demonstrate another example with a stoplight. *Ask students: What do you do if the light turns green? If the light turns yellow? If the light turns red?* Explain to students how they could use multiple if statements to demonstrate how a driver would make a decision depending on the color of the stoplight.

What is a Nested Statement?

A **nested statement** is any statement inside of another statement. This could mean a repeat loop inside of an if statement, an if statement inside of a repeat loop, an if statement inside another if statement, etc. You can have as many nested statements in a program as you'd like, but make sure that each statement has its appropriate start and end block! For example, when the program below is running, KIBO will shake every time it senses that it is near something.

CLEAN UP (5 min)

Using your classroom organizational system, have students clean up their work areas and put away all KIBO parts and blocks in their appropriate spots.

Lesson 11: Final Project - Writing the Wild Rumpus Composition

Powerful Idea From Computer Science:
Design Process

Powerful Idea From Literacy:
Writing Process

OVERVIEW

Students will begin their final project in this lesson: writing their Wild Rumpus composition. During the course of this final project, students will put to use all the concepts learned during the previous lessons. Students can use parts of their programs from previous lessons, but they should be encouraged to start fresh and transfer their skills to a new context.

PURPOSE

The purpose of the final project is to allow students to demonstrate the skills they have acquired throughout the previous lessons and to apply them in new, creative ways. By writing out their plans first before building with KIBO, students make purposeful decisions about their projects and understand that not all ideas on paper can transfer to the actual design.

ACTIVITIES

- Wild Rumpus Composition (20 min)
- Writing vs. Coding (10 min)
- Begin Designing Projects (20 min)
- Technology Circle (10 min)

STUDENTS WILL BE ABLE TO...

- Utilize the Writing Process by writing their Wild Rumpus composition
- Decide which of their ideas can and cannot be translated into KIBO programs

PREPARATION FOR TEACHERS

- ☐ Read through the Activity Guide
- ☐ Review the Lesson Slides
- ☐ Ensure all KIBO bodies have 4 working AA batteries
- ☐ Sort KIBO blocks and pieces (listed in Materials section) by part and place in a central location

MATERIALS

FOR THE TEACHER:

- Job Cards
- 1 flathead screwdriver
- Extra AA batteries
- Appendix E for troubleshooting tips

FOR STUDENTS:

- Design Journal (see Appendix C for example)
- Colored pencils or crayons
- KIBO bodies, wheels, motors, Begin and End blocks, blue Motion blocks, yellow Light blocks, Beep and Sing blocks, Wait for Clap blocks, Sound sensors, Sound Recorder module, orange Sound Recorder blocks, Repeat and End Repeat blocks, number parameters, Light and Distance sensors, gray sensor parameters, If and End if blocks, purple sensor parameters, platform pieces

Lesson 11: Activities

WILD RUMPUS COMPOSITION (20 min)

First ask students to take out their Design Journals and write down their top three activities they would have in their own Wild Rumpus. *You might first provide an example: In my Wild Rumpus, I would have an awesome dance party, some howling at the moon, and making s'mores. Ask students: What three things would you have in your Wild Rumpus?* Students should refer back to these three ideas throughout the final project.

Students will engage in the Writing Process to plan out their Wild Rumpus composition using their Design Journals. Based on their three ideas, explain to students they will write a story about their Wild Rumpus. Below are examples of things to include in their composition, as well as writing tips:

- Identify the audience and purpose of writing *(Who will be reading your ideas for your Wild Rumpus? What might they want to know about your project?)*
- Use prewriting strategies to generate ideas before writing Use organizational strategies to keep track of the different project components
- Organize writing to include a beginning, middle, and end *(How does the Wild Rumpus start, what happens during it, and how does it end?)*

Organize writing to include a beginning, middle, and end (How does the Wild Rumpus start, what happens during it, and how does it end?)

WRITING VS. CODING (10 min)

This activity provides a chance for students to reflect on the constraints and affordances of each medium, writing and coding. Have students come together in a technology circle. *Ask students:*

- *What activities did you include in your Wild Rumpus?*
- *Are there certain activities you wrote about that you can code with KIBO?*
- *Are there certain activities you wrote about that might not work with KIBO? How will you change your idea so that it makes sense for KIBO?*

BEGIN DESIGNING PROJECTS (20 min)

Explain to students: *Now that you have written about your Wild Rumpus, start planning your KIBO Wild Rumpus!"* Have students take out their Design Journals. Students will go through each step of the Design Process as they create their final KIBO projects. First, students will be asked: "*What activities do you want your robot to do in your Wild Rumpus?"* Students' responses should reflect any changes they decide to make after the class discussion on writing versus coding. Then students will be asked to imagine: "*What will your robot look like?"*. This provides an opportunity for students to begin imagining how they will decorate their KIBO with arts and crafts materials. Encourage students to use colored pencils or crayons so that they can fully imagine what their end product will look like.

TECHNOLOGY CIRCLE (10 min)

Come together in a technology circle. Have students share out their initial ideas for their KIBO projects. Ask students: "*What ideas do you have for your KIBO Wild Rumpus? How does that idea relate to what you had written in your Wild Rumpus Composition? What will your KIBO look like?*

Lesson 12: Final Project - Coding the Wild Rumpus (3 Day Plan)

Powerful Idea From Computer Science:	**Powerful Idea From Literacy:**
Design Process	Writing Process

OVERVIEW

In this final lesson, students will code their Wild Rumpus programs. During the course of this final project, students will put to use all the concepts learned during the previous lessons. When students are finished with their projects, they will share them with each other and offer their gratitude to those who have helped them along the way. Lesson 12 is structured to take three hours and should be spread out over multiple days (ideally 60 minutes over three days).

PURPOSE

The purpose of the final project is to allow students to demonstrate the skills they have acquired throughout the previous lessons and to apply them in new, creative ways. After students finish presenting their projects to their peers, educators are encouraged to invite families and community members to view students' final projects.

ACTIVITIES

- Day 1
  - Plan and Create (20 min)
  - Coding the Wild Rumpus (20 min)
  - Test and Improve (10 min)
  - Clean Up and Prep for Next Day (10 min)
- Day 2
  - Revise Projects (15 min)
  - Peer Feedback (10 min)
  - Collaboration Web (10 min)
  - Decorating KIBO (20 min)
  - Clean Up (5 min)
- Day 3
  - Prepare for Sharing (5 min)
  - Share Creations (20 min)
  - Clean Up (5 min)
  - Wild Rumpus Reflection (15 min)
  - KIBO Mastery Challenge D (15 min)

STUDENTS WILL BE ABLE TO...

- Demonstrate the Design Process in full by planning, designing, and creating a final KIBO project
- Share final projects with peers, family and community members
- Identify and show appreciation to those who have helped them with their final projects

PREPARATION FOR TEACHERS

- ☐ Read through the Activity Guide
- ☐ Review the Lesson Slides
- ☐ Rent the movie *Where the Wild Things Are* from the local library or movie rental kiosk (optional)
- ☐ Ensure all KIBO bodies have 4 working AA batteries
- ☐ Sort KIBO blocks and pieces (listed in Materials section) by part and place in a central location
- ☐ Print KIBO Mastery Challenge D (one for each student)

MATERIALS

FOR THE TEACHER:

- 1 copy of *Where the Wild Things Are* by Maurice Sendak
- Job Cards
- 1 flathead screwdriver
- Extra AA batteries
- Appendix E for troubleshooting tips

FOR STUDENTS:

- Design Journal (see Appendix C for example)
- Crafts and recycled materials
- Construction paper or other kind of paper to write thank you letters
- KIBO bodies, wheels, motors, Begin and End blocks, blue Motion blocks, yellow Light blocks, Beep and Sing blocks, Wait for Clap blocks, Sound sensors, Sound Recorder module, orange Sound Recorder blocks, Repeat and End Repeat blocks, number parameters, Light and Distance sensors, gray sensor parameters, If and End if blocks, purple sensor parameters, platform pieces

Lesson 12: Activities – Day 1

DAY 1:

PLAN AND CREATE (20 min)

Have students take out their Design Journals. Ask students: "*What KIBO parts and blocks will you need for your program? What kinds of arts and crafts materials will you use to decorate your KIBO?*" Have students use the KIBO Stickers to write what they think their program will be. Remind students that if their first program does not work the way they want to, that's okay! The point of this activity is to have some type of plan before starting to code the Wild Rumpus using KIBO. Encourage students to challenge themselves and make program plans that will use advanced parts such as sensors, Repeat loops, and If statements.

CODING THE WILD RUMPUS (20 min)

Once students have their plans in their Design Journals, they are ready to start building their program with KIBO! Take out KIBOs and blocks. Using their design plans, students will turn their compositional Wild Rumpus into a programmed Wild Rumpus.

By this point, students have had several opportunities to work in pairs or small groups. Feel free to continue using the Job Cards to enable collaboration and ensure that each student has the opportunity to be hands-on with KIBO.

TEST AND IMPROVE (10 min)

Explain to students that now that they have tested out their initial program with KIBO, they will evaluate, or test, to see what changes they may want to make to their projects. Have students take out their Design Journals and use the checklist to see how their robot is coming along. Read aloud each of the checklist questions and have students mark their progress by circling the happy face, neutral face, or sad face. Then have students reflect on the changes they will make to their program based on this feedback.

CLEAN UP AND PREPARE FOR NEXT DAY (10 min)

Using your classroom organizational system, have students clean up their work areas and put away all KIBO parts and blocks in their appropriate spots. If your classroom has enough space, designate an area of the room where students can save parts of their project. Explain to students that the next time, students will make changes to their projects and get feedback from their peers. Make sure to acknowledge students' persistence and effort as they spend time testing and improving their final KIBO projects - making a final project is hard work and requires time, patience, and feedback from others!

Lesson 12: Activities – Day 2

DAY 2:

REVISE PROJECTS (15 min)

Now that students have ideas for how they will make changes to their KIBO project, have students take out KIBOs and incorporate those changes!

PEER FEEDBACK (10 min)

Split students into pairs. One student (Student A) will share their revised project to their partner (Student B) for 3 minutes. Then Student B will give feedback for 2 minutes. Then switch! Encourage students to say one thing they liked about their partner's project and one suggestion for how to improve their partner's project.

COLLABORATION WEB (10 min)

A collaboration web is a tool used to foster collaboration and support. Each child receives a printout with their photograph in the center of the page and the names and photographs of all the other children in the class arranged in a circle surrounding the central photo. Prepare these printouts ahead of time.

Pass out each student's Collaboration Web. Have students draw lines on their Collaboration Webs from their picture in the middle to pictures of other students in the class who gave them good pieces of feedback or helped in a different way. Encourage students to continue filling out their Webs as they continue with their final projects.

DECORATING KIBO (20 min)

Have students make their Wild Rumpus come to life using arts and crafts materials to decorate their KIBO! Refer back to the KIBO resources on how to secure decorations onto KIBO using the different art platforms. Students should also take this time to incorporate any changes to their project based on their peer feedback.

CLEAN UP (5 min)

Using your classroom organizational system, have students clean up their work areas and put away all KIBO parts and blocks in their appropriate spots.

Lesson 12: Activities – Day 3

DAY 3:

PREPARE FOR SHARING (5 min)

Ask students to test their final program one last time individually so that it is ready for sharing! If students took apart their codes after the last class, have students take this time to reassemble their program and secure their decorations onto KIBO.

SHARE CREATIONS (20 min)

During the final presentations, have students present their Wild Rumpus compositions and KIBO projects. Students can share their final projects altogether in a technology circle, or as a gallery walk, in which half of the students walk around the classroom to each project while the other half present their projects. Then the two groups switch. Students should share:

- their robots and decorations
- why they chose the features they did for their robot
- the final program and what each block and module represent
- anything that was hard, easy, surprising, interesting, etc. about the process
- which peers did they collaborate with the most (based on their Collaboration Web) and want to thank personally (e.g., with a high five, thank you card, etc.)

Important! Before disassembling students' final projects, take pictures of each decorated KIBO robot and program. Instructions for taking pictures of final projects are included in the Lesson Slides.

CLEAN UP (5 min)

Using your classroom organizational system, have students clean up their work areas and put away all KIBO parts and blocks in their appropriate spots.

WILD RUMPUS REFLECTION (15 min)

Have students take out their Design Journals and answer the four reflection questions. Then have students complete the Student Perceptions Survey. Explain to students that they should be as honest as possible when answering how they felt doing these activities. As students are reflecting in their Design Journals, take pictures of students' final projects. Pictures should include both the decorated KIBO and the full code.

KIBO MASTERY CHALLENGE D (15 min)

On the Appendix B-2 you will find challenge D. Please hand out one copy of the assessment to each child in your class.
Instructions:

- Read each question and option out loud to the group. Students can ask to have questions or options read out loud up to 3 times.
- Instruct children to circle only 1 answer per question.
- Make sure students answer the questions by themselves. Students should not be discussing or copying answers.
- Hand in completed answer sheets to Angela de Mik or a member of your school's assessment team.

Appendix A. Materials

Appendix A. Materials

Robotics and Technology Materials:

- ☐ KIBO 21 robotics kits: It is recommended to deconstruct the kits into individual parts or to provide each pair of students with one pre-made kit, so that students can access only the parts required for each lesson.
- ☐ 1 flathead screwdriver
- ☐ AA batteries (each KIBO kit requires 4 AA batteries)
- ☐ Extra AA batteries
- ☐ Flashlights (one per pair of students)

Art and Game Materials:

- ☐ Construction paper or other kind of decorative paper
- ☐ Crafts and recycled materials (e.g. scrap paper, scissors, straws, popsicle sticks, recycled cardboard, any other available materials that students can use to decorate their robots)
- ☐ Masking tape

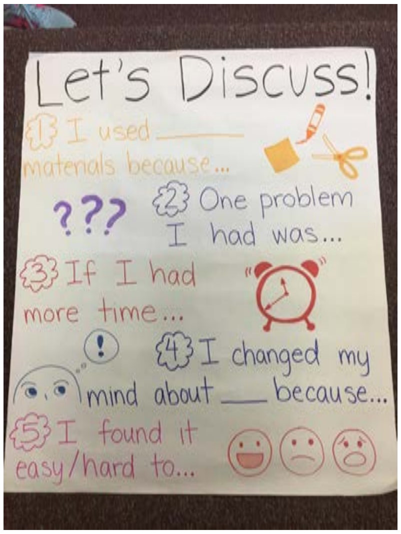
Teaching Materials:

- ☐ 1 copy of *Where the Wild Things Are* by Maurice Sendak (it might be helpful to have multiple copies for students to reference during projects)
- ☐ 1 copy of *My Five Senses* by Aliki
- ☐ Large KIBO Says cards (purchase from KinderLab Robotics or make your own)
- ☐ Anchor charts **(see following pages for examples or use Lesson Slides)**
    - Discussion sentence starters
    - Design Process
    - Writing Process
    - KIBO Robot Parts song
    - Characteristics of Robots comparison chart
    - Why is KIBO Confused?
- ☐ Printed pictures **(see following pages for examples or use Lesson Slides)**
    - 8-10 pictures of naturally occurring and man-made objects
    - 8-10 pictures of robots and non-robots
    - Large letter cards: A, R, C
    - Handwritten and typed message

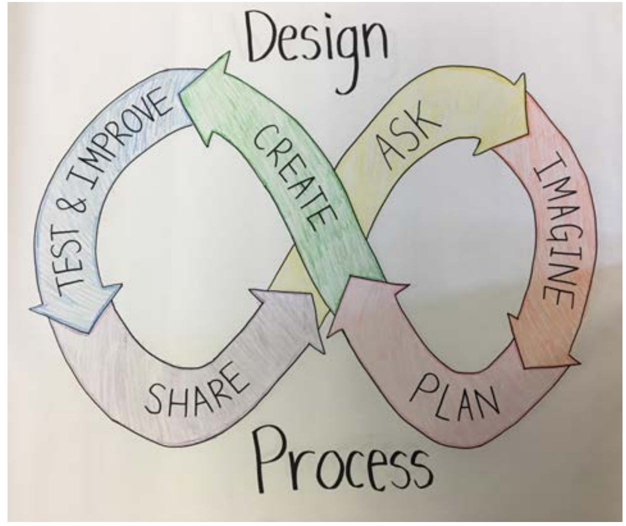
Optional Materials:

- ☐ Small KIBO block cutouts (download from http://bit.ly/KIBOcutouts)
- ☐ KIBO Parts Bingo cards (download from http://bit.ly/KIBOpartBINGO)
- ☐ KIBO Blocks Bingo cards (download from http://bit.ly/KIBOblockBINGO)

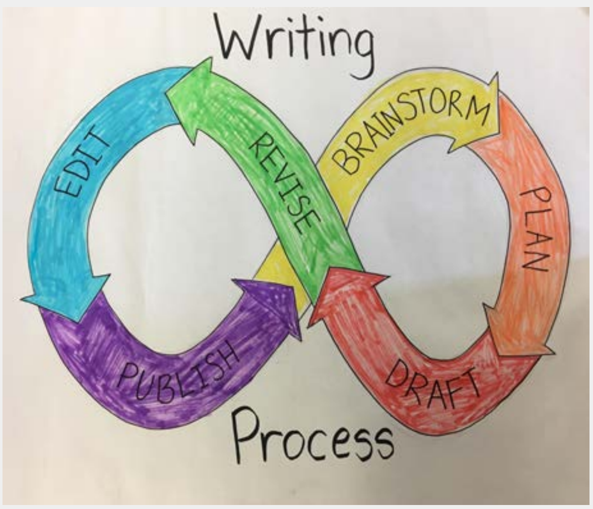
Discussion Sentence Starters



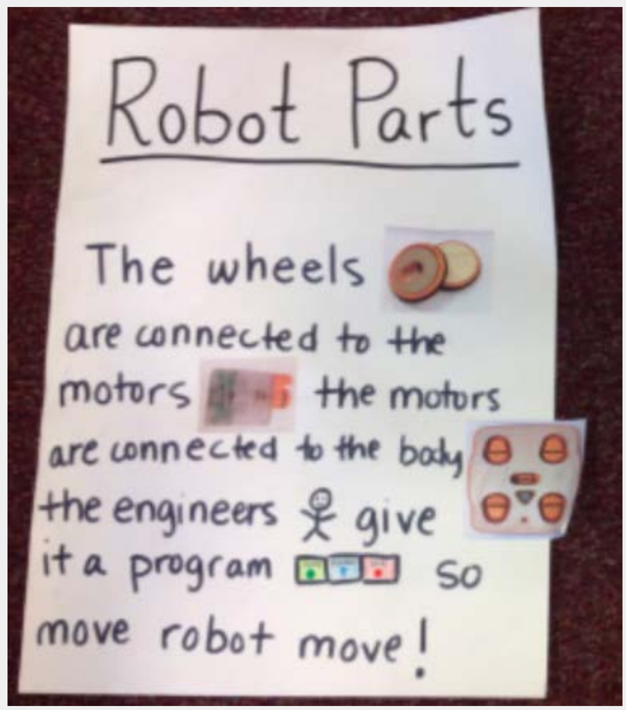
Design Process



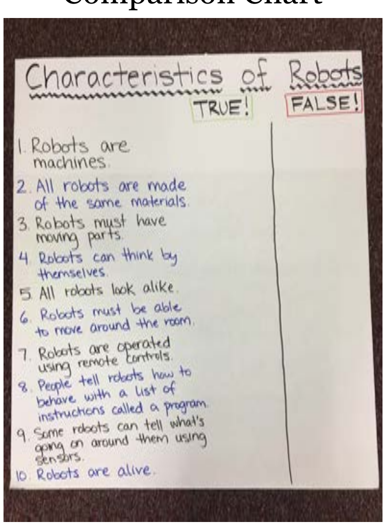
Writing Process



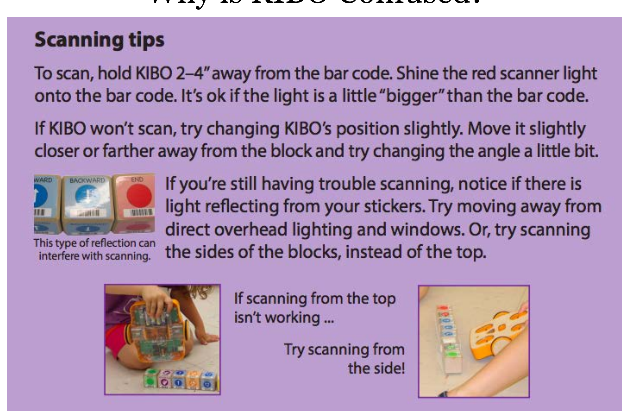
KIBO Robot Parts Song



Characteristics of Robots Comparison Chart



Why is KIBO Confused?



Scanning tips

To scan, hold KIBO 2–4" away from the bar code. Shine the red scanner light onto the bar code. It's ok if the light is a little "bigger" than the bar code.

If KIBO won't scan, try changing KIBO's position slightly. Move it slightly closer or farther away from the block and try changing the angle a little bit.

If you're still having trouble scanning, notice if there is light reflecting from your stickers. Try moving away from direct overhead lighting and windows. Or, try scanning the sides of the blocks, instead of the top.

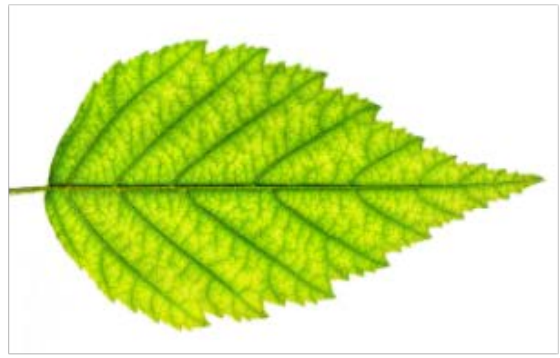
This type of reflection can interfere with scanning.

If scanning from the top isn't working ...

Try scanning from the side!

Examples of Printed Pictures:

8-10 pictures of naturally occurring and man-made objects



Examples of Printed Pictures:

8-10 pictures of robots and non-robots

For non-robot pictures: reuse the images from the man-made objects in Lesson 1

Example of Typed/Printed Message

Molly went to the store to buy apples and crackers.

Appendix B-1. KIBO Mastery Challenge A

KMC A

Name ___________________________ Date _________________

Teacher ________________________

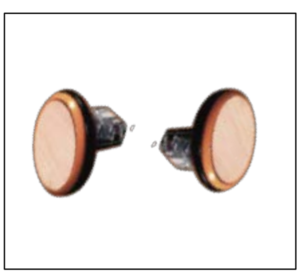
Circle the correct answers:

1. Which part of the block does KIBO read?

2. Which block makes KIBO shake?

3. You want KIBO to turn its light on, but KIBO is not turning its light on!

What does KIBO need for it to turn its light on?




KMC
A

Name ______________________ Date ______________

Teacher ____________________

4. Only **one** of these programs works. Which program works?

KMC A

Name ________________________ Date _______________

Teacher _____________________

5. KIBO moved forward and then started shaking. Why did KIBO do that?

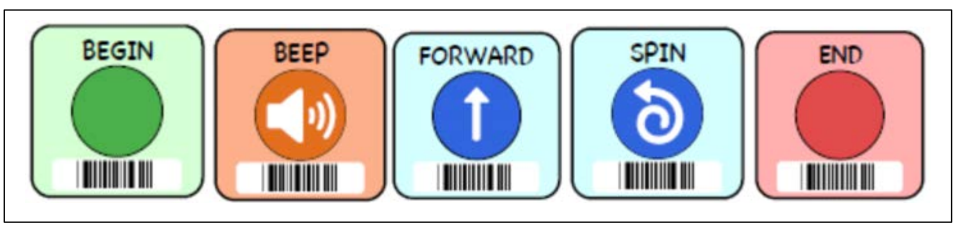
KIBO was scared
The programmer programed KIBO to do that
KIBO did not want to keep moving forward
KIBO wanted to dance

KMC
A

Name ______________________ Date ______________

Teacher ____________________

6. How can you make your KIBO spin, then beep, and then move forward?



Appendix B-2. KIBO Mastery Challenge B

KMC
B

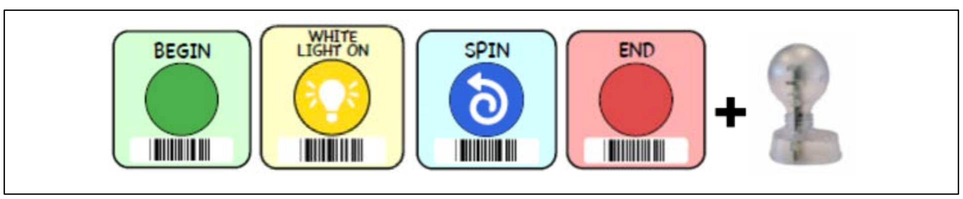
Name ________________________ Date _______________

Teacher _____________________

Circle the correct answers:

1. This program makes KIBO shake after hearing a clap.

Which program will make KIBO spin after hearing a clap?



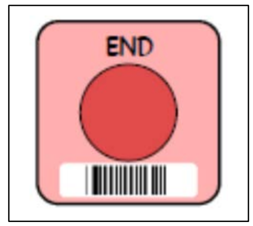
KMC
B

Name ________________________ Date _______________

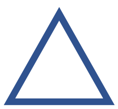
Teacher _____________________

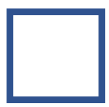
2. This is your program:

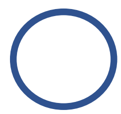
Which block is in the wrong place?



3. In class, you recorded these sounds with the sound recorder.


"Hello"


"Wow"


"Yay!"

Which program will make KIBO say "Wow", spin around, and then say "Yay!"?

Name ________________________ Date _______________

Teacher _____________________

4. Imani is having trouble with programming her KIBO and asked for your help!

   She wants her KIBO to move forward after it hears a clap, and then turn a white light on. This is what she has now:

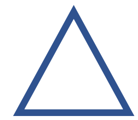
   How can Imani change it to make KIBO do what she wants?

KMC
B

Name ________________________ Date _______________

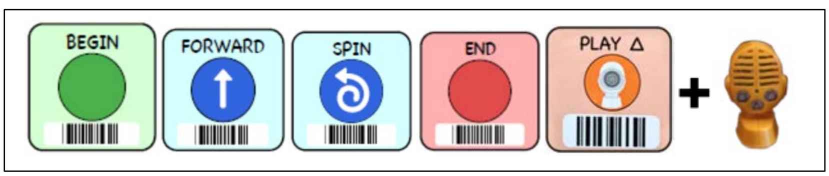
Teacher _____________________

5. It's Jordan's birthday, help him celebrate!



This is what's recorded on KIBO's sound recorder: "Happy birthday!"

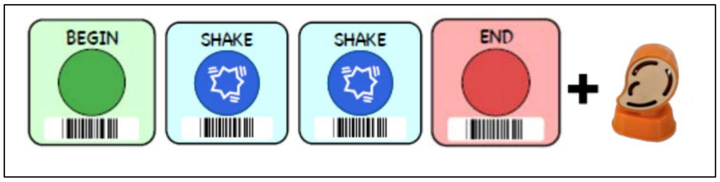
How can you **best** program KIBO to help celebrate?

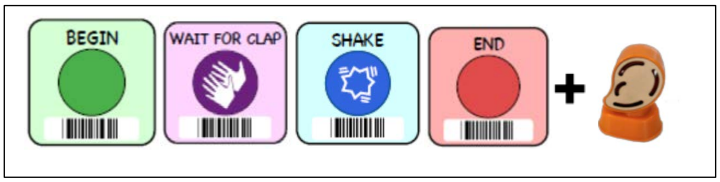


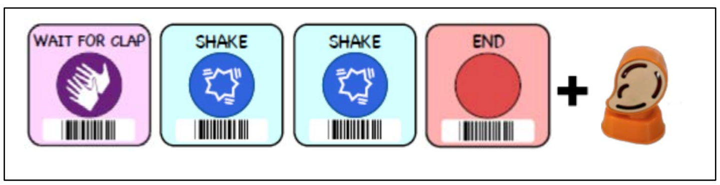
KMC B Name ________________________ Date _______________

Teacher _____________________

6. Which program makes KIBO shake, but only if you clap?







Appendix B-3. KIBO Mastery Challenge C

KMC
C

Name________________________ Date _______________

Teacher ______________________

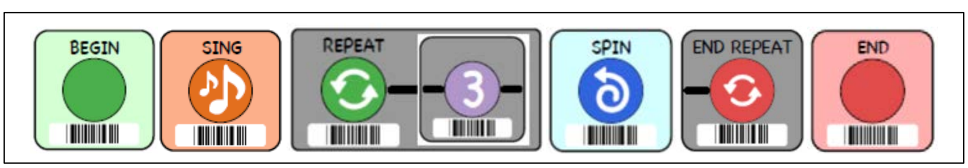
Circle the correct answers:

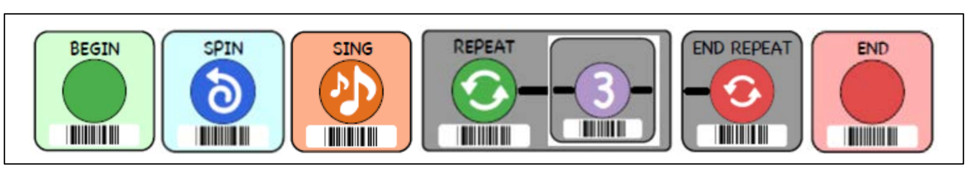
1. Miriam wrote this program for her KIBO:

Which part is **NOT** needed?

2. Before you learned about repeat loops, you wrote this program:

How would you write this program using repeat loops?



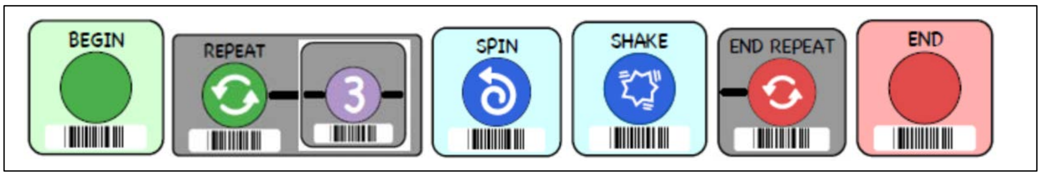


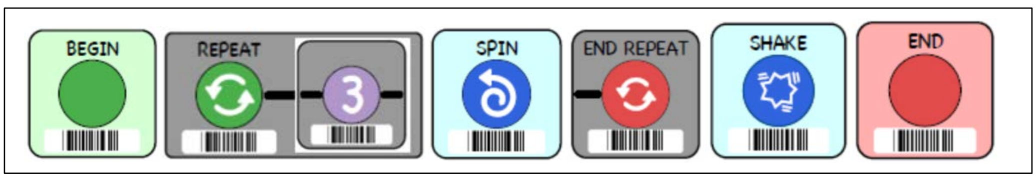
Name_______________________ Date _______________

Teacher _____________________

3. You want KIBO to spin three times and shake once, but your program doesn't do what you want.

How can you change your program to make KIBO spin three times and shake once?





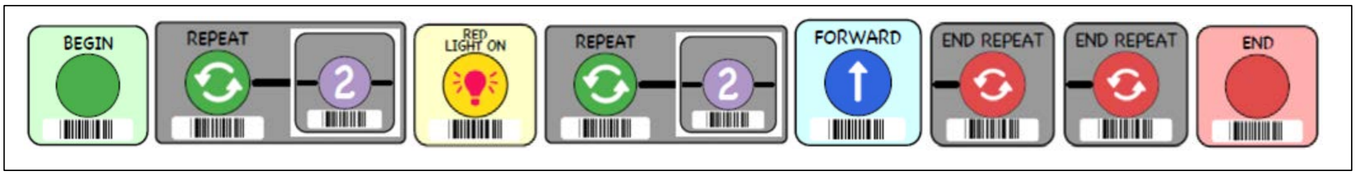
KMC C

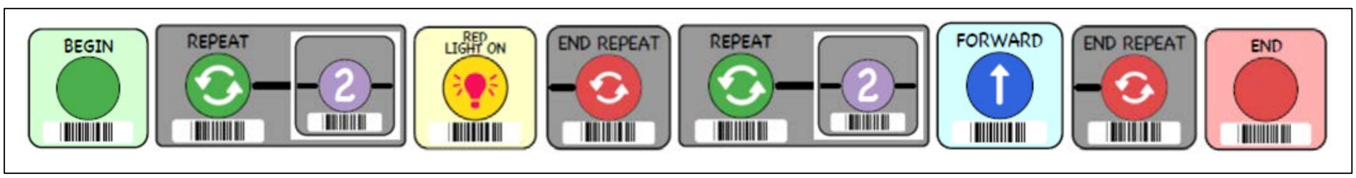
Name_______________________ Date _______________

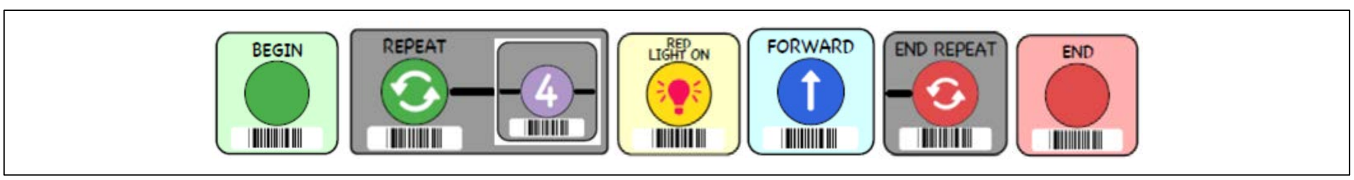
Teacher _____________________

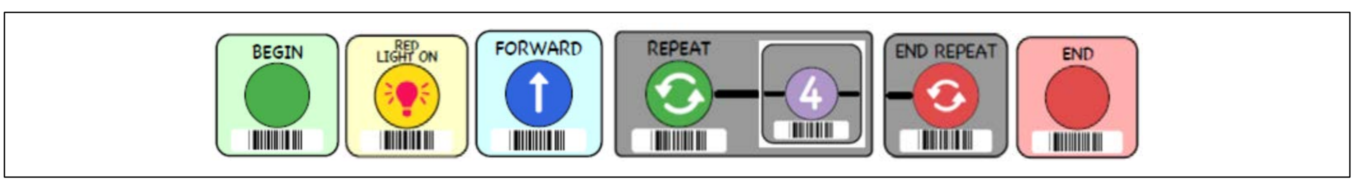
4. Here's another program you wrote before you learned about repeat loops:

How would you write this program using repeat loops?





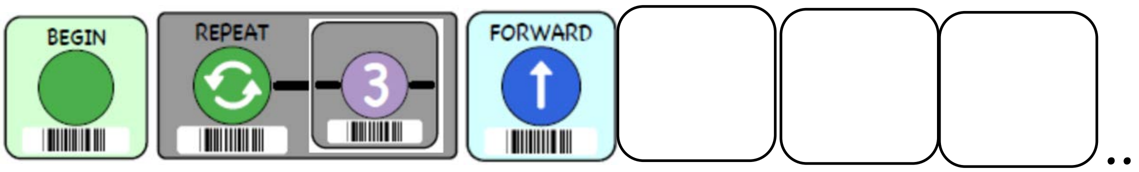




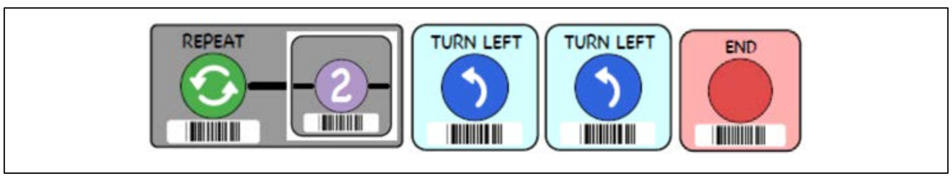
Name________________________ Date _______________

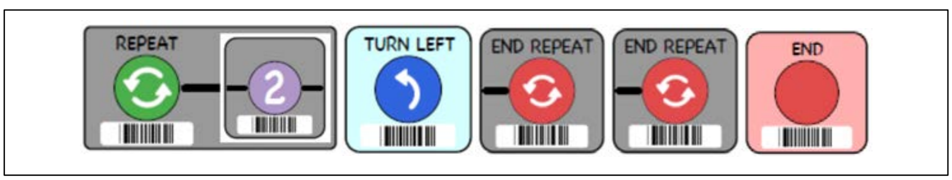
Teacher ______________________

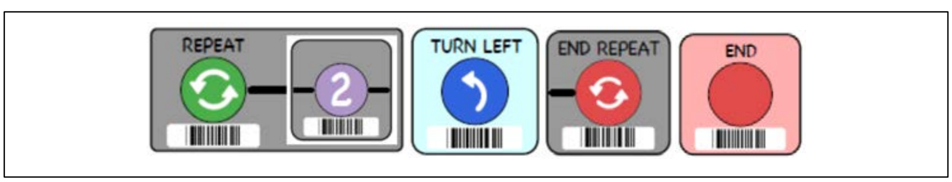
5. You want your KIBO to move forward three times, then turn left twice.
This is how you started your program so far:



What blocks should you add to finish it?







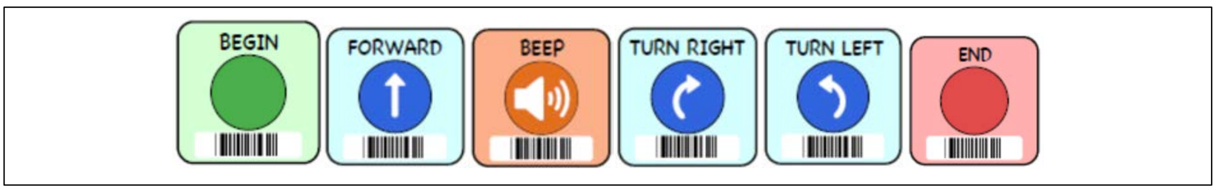
Name________________________ Date ______________

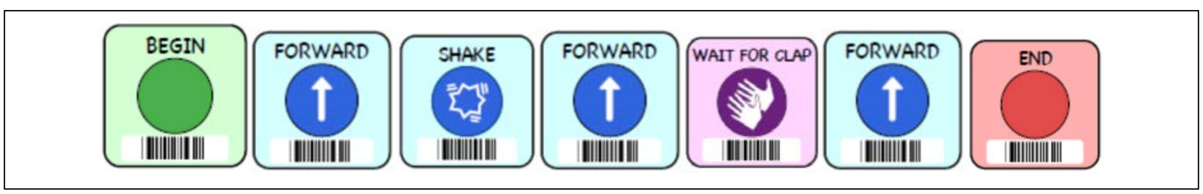
Teacher ____________________

6. Hasan wrote a story to match his program. Here is his story:

   "One day, KIBO left its house and was walking down the street when it heard a loud boom of thunder. KIBO started to shake because it was so scared of the thunder. It quickly moved backwards to the house."

   Which program **best** matches his story?





Appendix B-4. KIBO Mastery Challenge D

KMC
D

Name________________________ Date _______________

Teacher ______________________

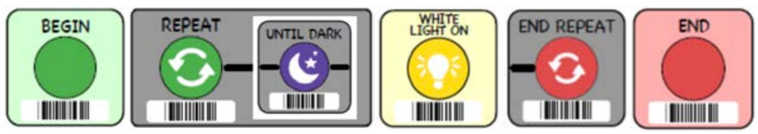
Circle the correct answers:

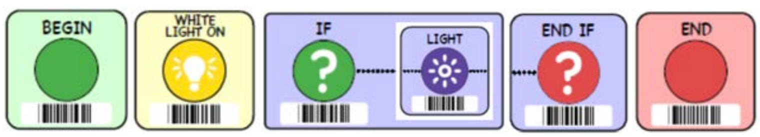
1. You want to make KIBO sing and beep! You want KIBO to always beep, but only sing if it is far from something. How can you **best** program this?

Name________________________ Date _______________

Teacher ______________________

2. You want KIBO to be a nightlight. When it is dark outside, you want KIBO to keep its light on. When it is light outside, you want KIBO to turn its light off. How can you **best** program this?





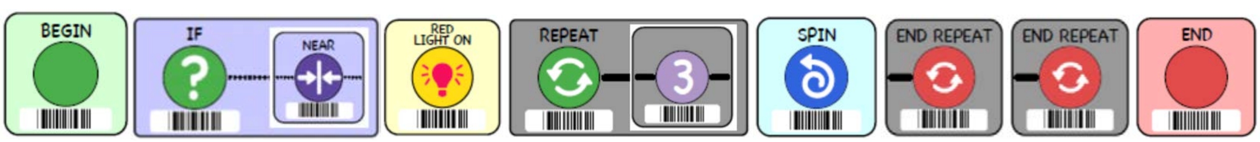
KMCD

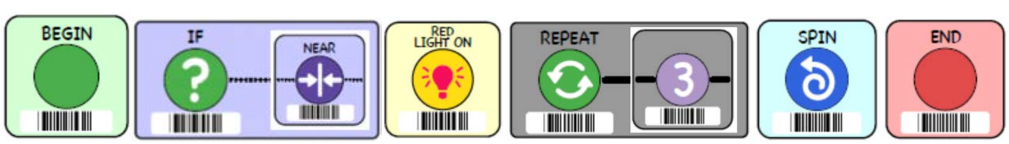
Name________________________ Date _______________

Teacher ______________________

3. You are programming KIBO to turn its red light on if it is near something. Then, you want KIBO to spin three times. This is the program you have so far, but it doesn't work:

How can you change it to make KIBO do what you want?





KMC
D

Name________________________ Date _______________

Teacher ______________________

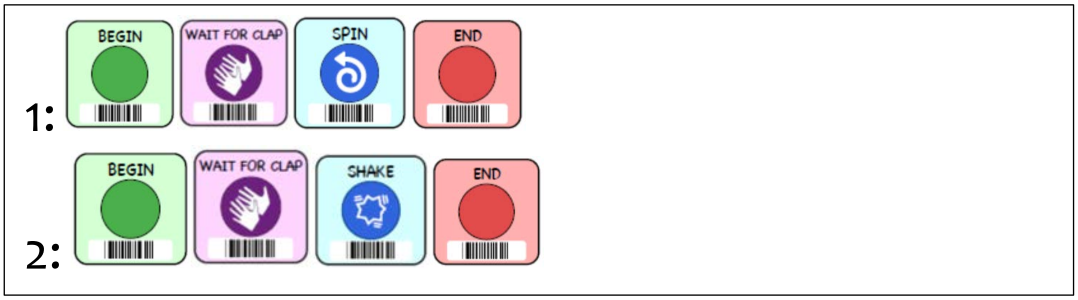
4. You made a program for KIBO. If KIBO is in the dark and **NEAR** something, then it will turn its white light on and shake. If KIBO is in the dark, but **NOT NEAR** something, then it will only shake.

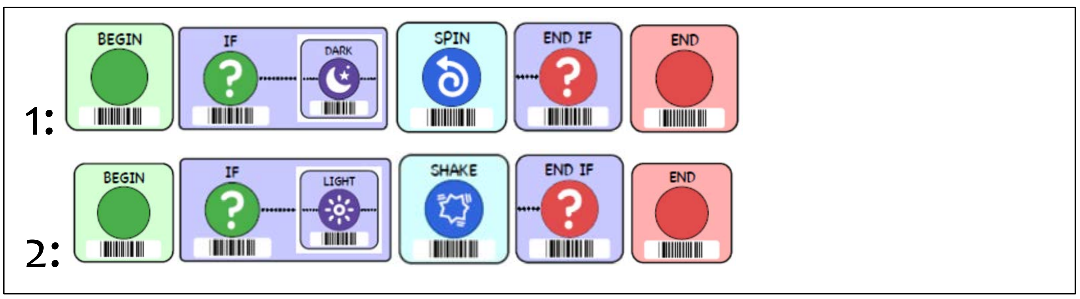
Which of these is your program?

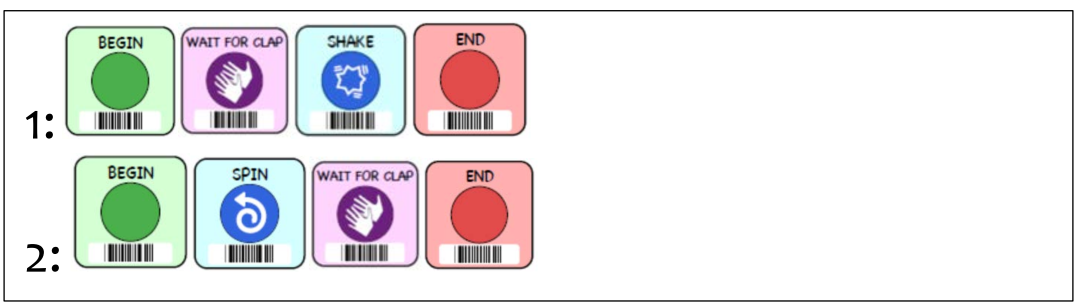
Name_______________________ Date _______________

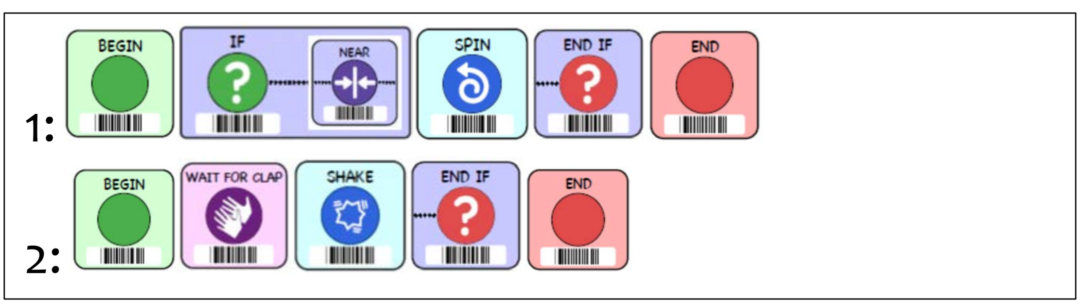
Teacher _____________________

5. I have KIBO 1 and you have KIBO 2. We want them to dance together. How can we **best** program them to start dancing at the **same time**?









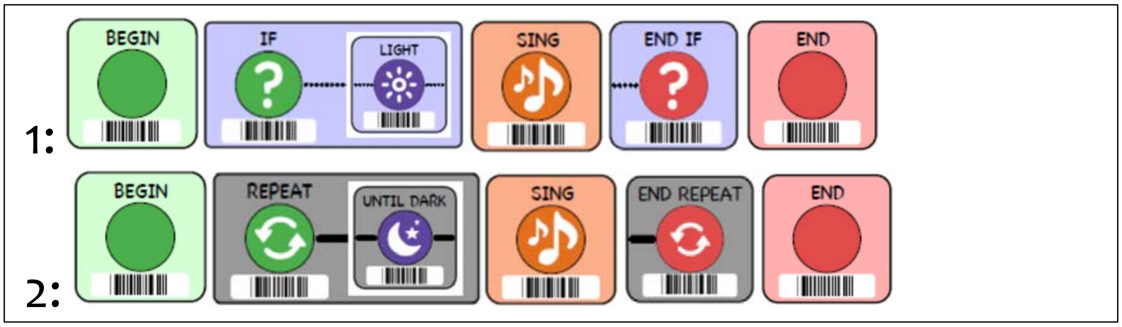
KMCD

Name________________________ Date _______________

Teacher ______________________

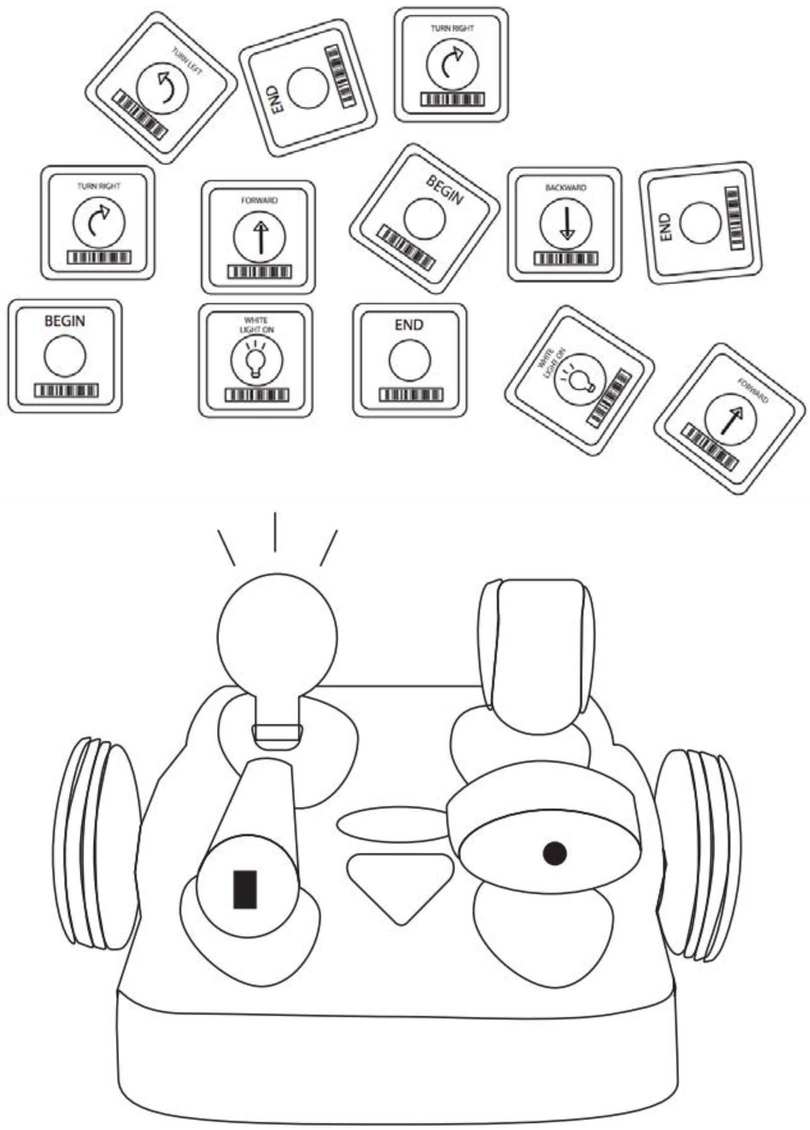
6. Now we want the KIBOs to sing together! Three of these programs will make them start singing at the same time, and one will not.

   Which one does **NOT** make KIBO 1 and KIBO 2 start singing at the same time?



Appendix C. Design Journal

____________’s Design Journal



Name: ____________________ Date: ____________________

Lesson 1:
How to Make a Pizza

Draw a picture of your pizza.

What are the steps for making a pizza?

____

____

____

Name: ____________________ Date: _____________________

Lesson 1:
How to Make a Pizza (continued)

What toppings will you put on your pizza?

____

____

____

____

In what order will you put the toppings on your pizza?

____

____

____

____

____

Name: ____________________ Date: _____________________

Lesson 3:
Writing Reflection

What if the first scene was Max on a boat? How would that change the story?

____

____

____

____

____

Write a new opening to the story with Max on a boat. How did Max get on the boat?

____

____

____

____

____

Name: ____________________ Date: ____________________

Lesson 4:

Hokey-Pokey Reflection

Use the KIBO stickers to write your Hokey-Pokey program here.
Make sure the blocks are in the right order!

What is your favorite block in your program?

____

What does the block do?

____

How did the block help you program the Hokey-Pokey?

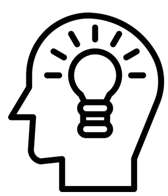
____

Name: ____________________ Date: _____________________

Lesson 5:
Debugging Reflection

What was one problem you had with KIBO? 

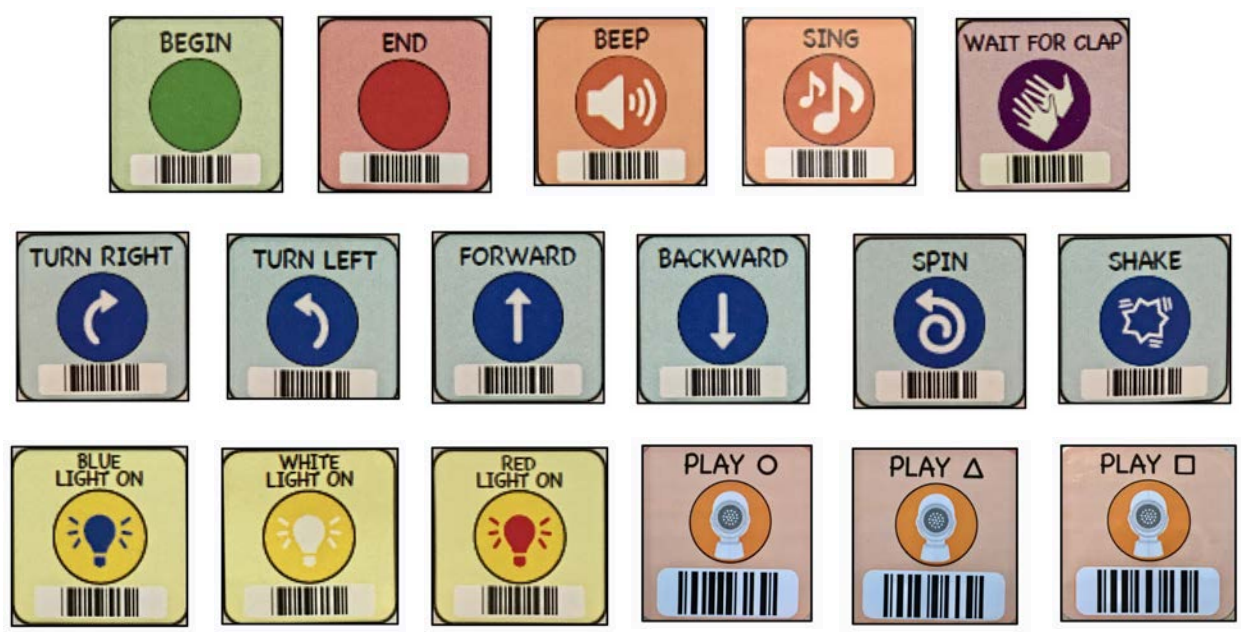
What were some things you tried to help solve the problem?

Which solution worked best? 

Name: ____________________ Date: _____________________

Lesson 7:
If You're Wild and You Know It

What blocks and modules would you use to program your robot to be happy and wild? Circle them:



What was your favorite block in your program?

____

What does this block do?

____

How did the block help you program KIBO to be happy and wild?

____

Name: ____________________ Date: _____________________

Lesson 7:

If You're Wild and You Know It (continued)

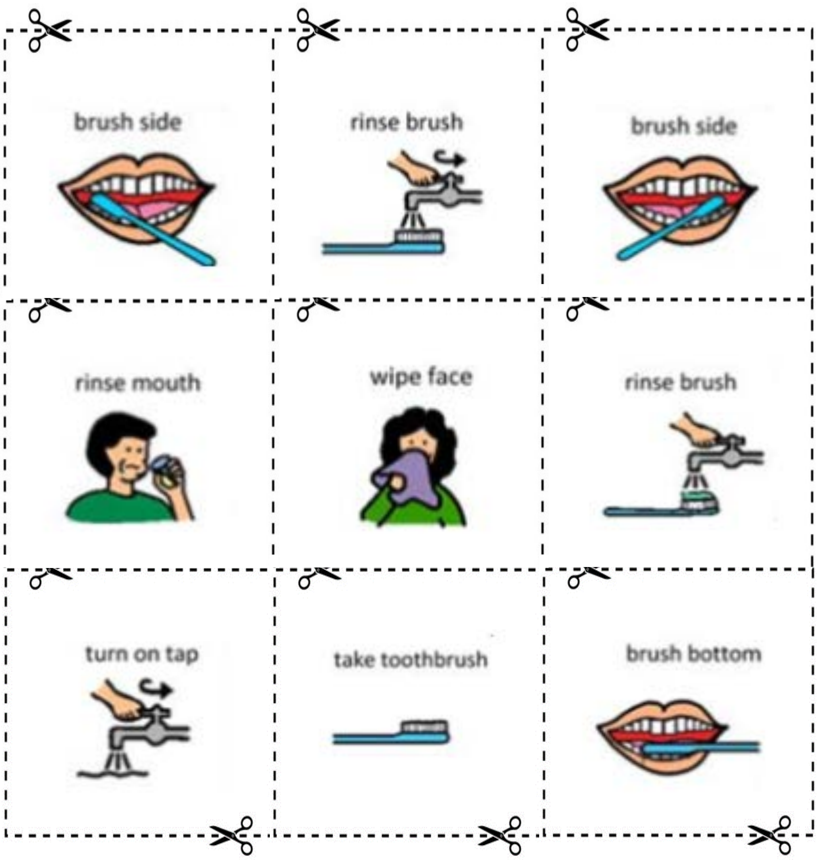
Use the KIBO stickers to write the blocks of your final program here:

Name: ____________________ Date: _____________________

Lesson 8:

Toothbrush Sequence Pictures

Cut out the pictures and paste them on the next page.

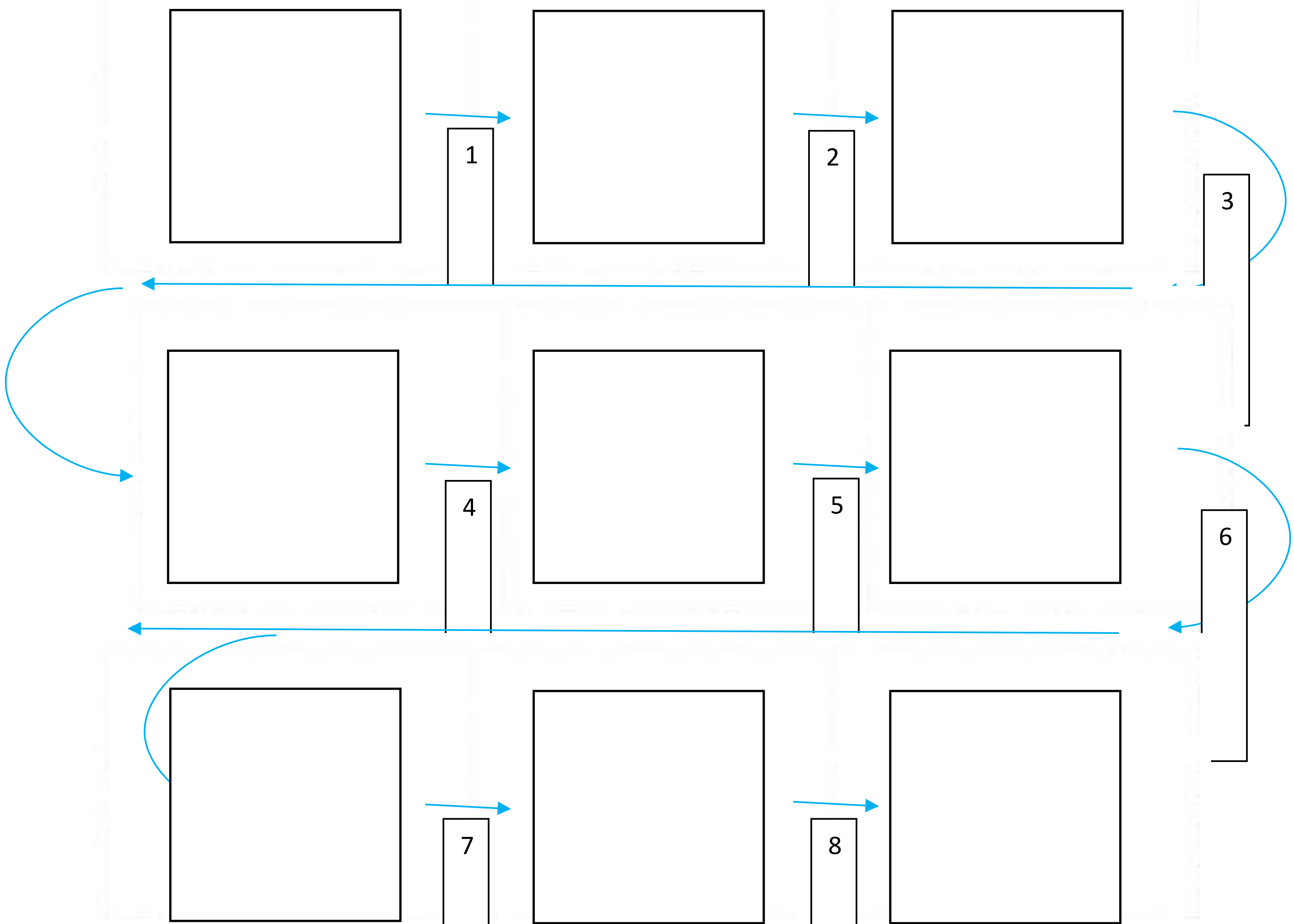


Name: ____________________ Date: _____________________

Lesson 8:

Toothbrush Exercise

Cut and paste the toothbrush sequence pictures in order:



Are there any steps missing?

____

____

Then circle the numbers where the new steps should go.

Name: ____________________ Date: _____________________

Lesson 10:
Writing an Alternative Story

Write your own ending to *Where the Wild Things Are*! What would you want to happen differently in the story?

Use the space below to draft your alternative story:

____

____

____

____

____

____

____

____

____

____

____

____

Name: ____________________ Date: _____________________

Lesson 11:
Wild Rumpus Composition

Describe your Wild Rumpus on the lines below. Make sure to add details!

___

___

___

___

___

___

___

___

___

___

___

Name: ____________________ Date: _____________________

Lesson 11:
Begin Designing Projects

Now that you have written about your Wild Rumpus, start planning your KIBO Wild Rumpus!

Ask: What actions do you want your robot to do in your Wild Rumpus?

____

____

____

____

Imagine: What will your robot look like?

My robot is a(n) ______________________________________

Its name is ____

Draw what your robot will look like:

Name: ____________________ Date: _____________________

Lesson 12:

Day 1: Plan and Create

Plan: Which KIBO parts and blocks will you need to create your project?

Circle the KIBO parts you think you will need:

Circle the programming blocks you think your robot's program will need:

Name: ____________________ Date: _____________________

Create: Gather your materials and programming blocks and get to work, engineer!

Describe the kinds of materials you used to decorate your KIBO:

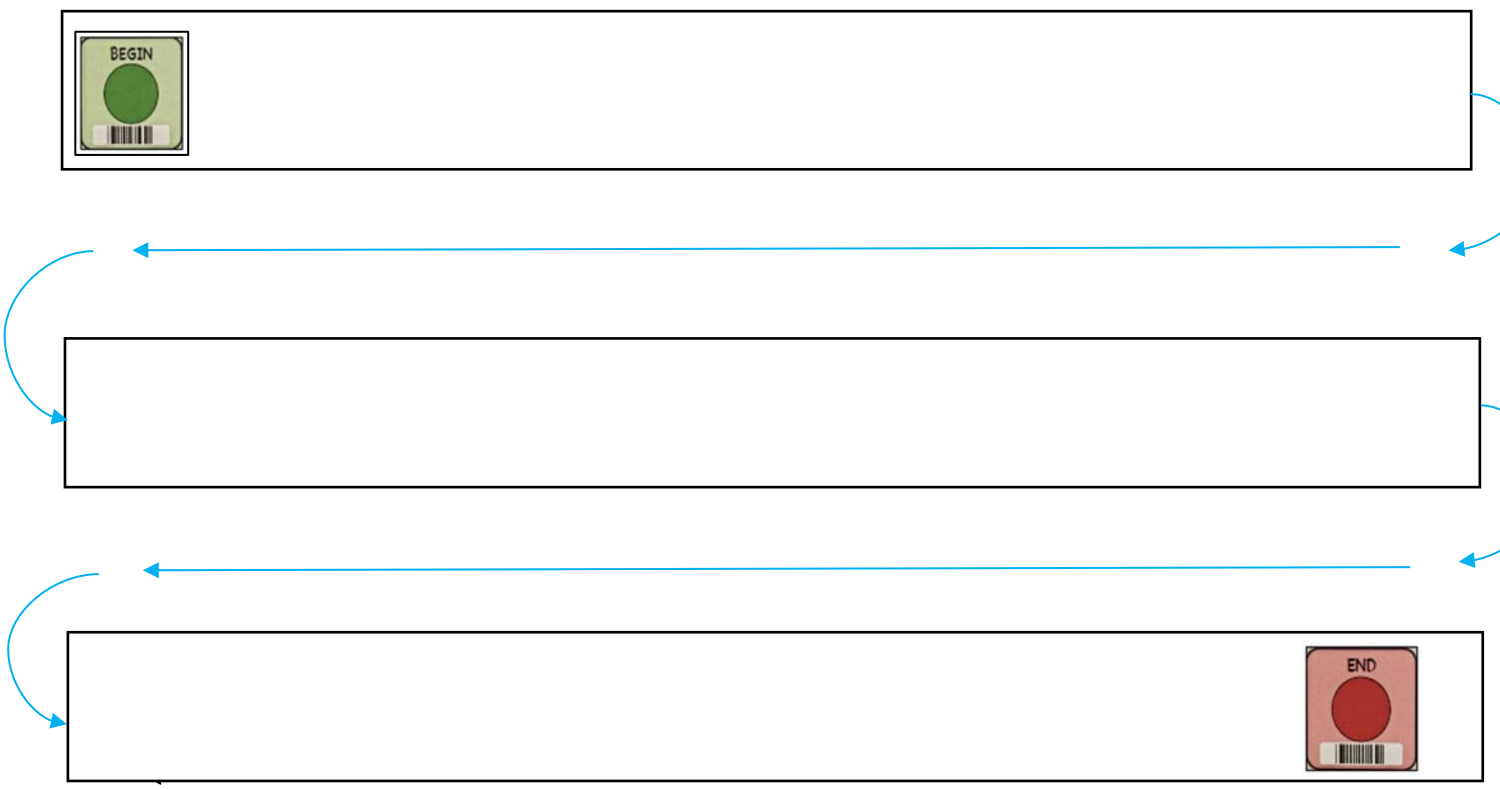
____

____

____

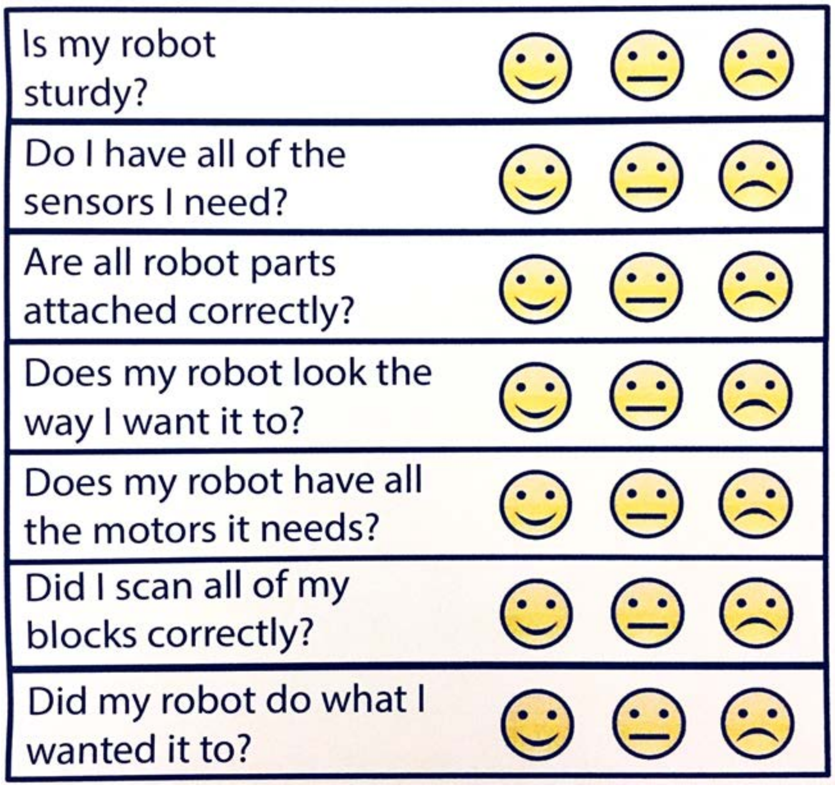
____

Use the KIBO stickers to write your robot's first program here. Don't worry if your first program doesn't work the way you want it to. You can always go back and make changes!



Name: ____________________ Date: _____________________

Test and Improve: Before engineers finish a project, they need to test and improve their work. Use this checklist to see how your robot is coming along!



Is my robot sturdy?	🙂	😐	🙁
Do I have all of the sensors I need?	🙂	😐	🙁
Are all robot parts attached correctly?	🙂	😐	🙁
Does my robot look the way I want it to?	🙂	😐	🙁
Does my robot have all the motors it needs?	🙂	😐	🙁
Did I scan all of my blocks correctly?	🙂	😐	🙁
Did my robot do what I wanted it to?	🙂	😐	🙁

Now it's time to improve and fix your "bugs"! What changes did you make to your project?

____

____

____

Name: ____________________ Date: _____________________

Lesson 12:

Day 2: Revise Projects

Use the KIBO stickers to record your robot's final program.

Name: ____________________ Date: _____________________

Lesson 12:

Day 3: Wild Rumpus Reflection

Share: Now that you have completed your robot, you can share what you have learned with others!

What was your favorite block in your program?

____

____

____

What does this block do?

____

____

____

How did the block help you program KIBO's Wild Rumpus?

____

____

____

Name: ____________________ Date: _____________________

Use this blank sheet for any extra planning or writing!

Name: ____________________ Date: _____________________

Use this blank sheet for any extra planning or writing!

Name: ____________________ Date: _____________________

Use this blank sheet for any extra planning or writing!

Appendix D. Collaboration Web

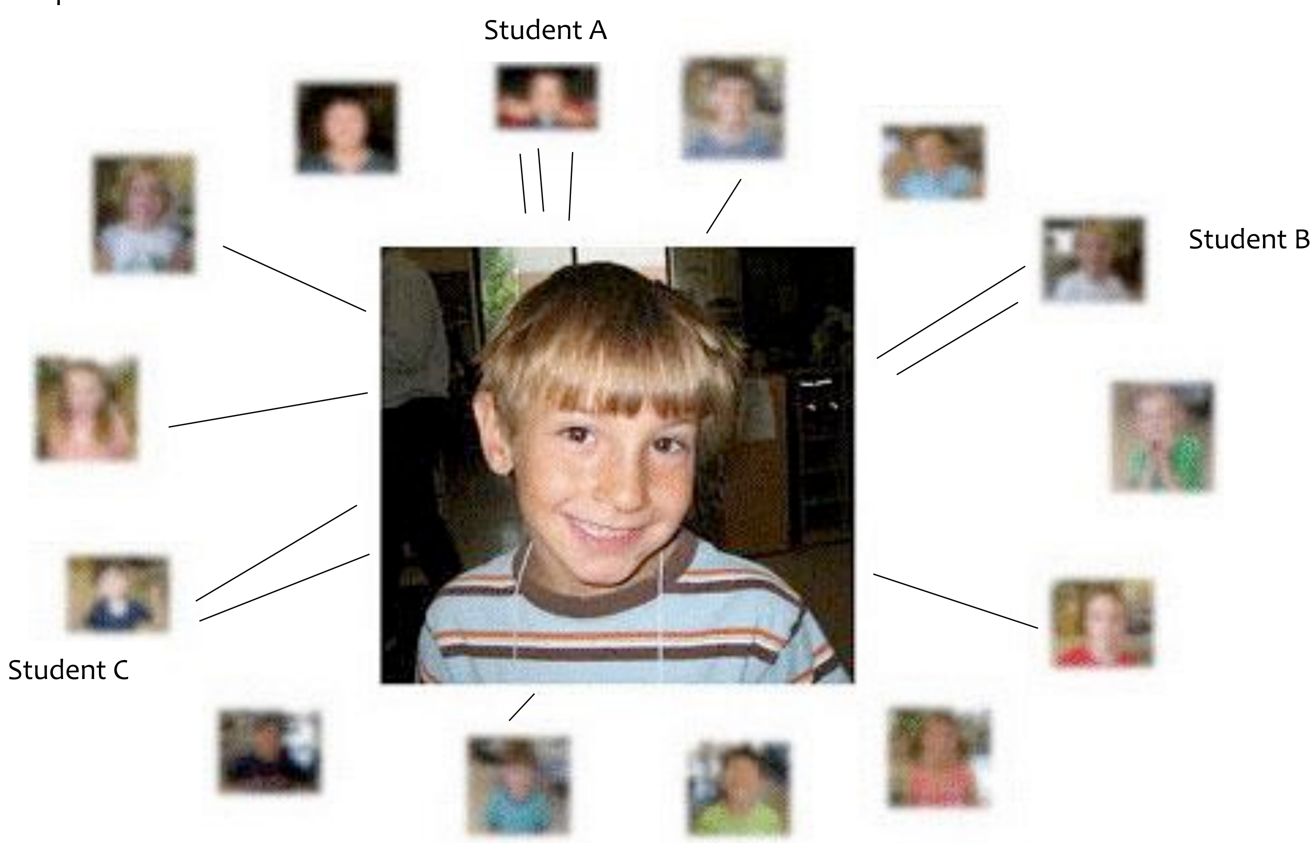
Appendix D. Collaboration Web

A collaboration web is a tool for students to recognize peers who have helped and supported them in different ways, such as working together on a common task, lending or borrowing materials, programming together, etc. Students will create a Collaboration Web during Lesson 8: The Wild Rumpus Project and write thank you letters to the three peers with whom they have collaborated the most.

Directions:

1. Obtain headshots of each student in the class.
2. Create individual printouts with each student's photograph in the center of the page and the names and photographs of all the other students arranged in a circle surrounding the central photo.
3. Whenever you observe students collaborating during the final project, ask students to draw a line from their photo in the center to the photo of the other students with whom they collaborated.
4. At the end of Lesson 8, ask students to count the number of lines they have with each student. Ask students to write thank you letters to the three students who have the most lines drawn to their photos.

Sample Collaboration Web:



The student in the center will write thank you letters to Students A, B, and C.

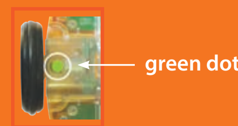
Appendix E. KinderLab's Troubleshooting Tips

Getting started with KIBO

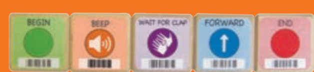
1. If this is your first time using your KIBO, insert 4 AA batteries into the battery case. The red scanner light will start blinking.

Screwdriver and batteries not included

2. Choose the motors, wheels, and sensors that you want to use. Insert the motors so that the green dot shows through KIBO's transparent bottom.



3. Sequence some blocks into a program. Every program needs a BEGIN block and an END block.



4. Push KIBO's triangular button to turn KIBO on. The red scanner light will blink.

KIBO will turn itself off if left alone for a few minutes.

5. Use KIBO to scan the bar codes on the programming blocks, left to right, one at a time*. If your scan was successful, KIBO will beep and the scan indicator LED will glow green after each block.

(A red scan indicator LED indicates a scanning error.)

**See scanning tips on the back of this guide.*

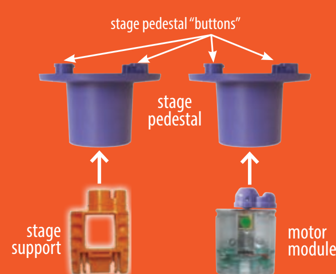
6. Push KIBO's triangular button to tell KIBO to go!

To re-run the program: push KIBO's button again.

To change your program: re-arrange the blocks, re-scan, and push KIBO's button. Watch KIBO go!

7. Decorate KIBO with the round or rectangular stage.

Insert stage support or motor module into stage pedestal



Insert the wood stage onto the stage pedestal buttons

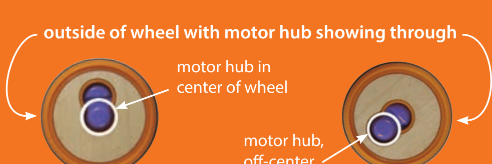
Insert stage assembly into middle motor socket

Decorate and play!

Fun things to try

Try inserting the motors "upside-down," with the green dot **not** showing and see what happens.

Insert the motors into the wheels so that the motors' axles are ***off-center***, relative to the center of the wheels, and see what happens!



Check out more fun challenges and activity guides at **http://resources.kinderlabrobotics.com.**

Good things to know

KIBO's lights can tell you lots of useful things:

◦ KIBO's red scanner light and triangular button will blink when KIBO is ready to ***scan*** a program – OR – when KIBO is ready to ***run*** a program. The button will stop blinking while KIBO is scanning a program; the red scanner light will stop blinking while KIBO is running a program.

◦ When KIBO's triangular button blinks, it means that KIBO has a program stored in its memory. The triangular button will go dark while KIBO is scanning a new program, and also after inserting new batteries.

◦ You can put KIBO to sleep by pressing and holding the triangular button for several seconds.

KinderLab robotics

Tips & Troubleshooting

Uh-oh ...

If the red scanner light is not blinking, it usually indicates a problem with the batteries. Remove and re-install the batteries. If that doesn't help, replace the batteries with new ones.

A tri-tone sound and a red scan LED means that an error occurred. KIBO may have mis-scanned, or there may be an error in your program. Try scanning again or re-arranging your blocks. Have fun experimenting!

If KIBO is turning left or right (or going backward) when it should be going forward, check the motors to make sure that the green dots are showing through KIBO's transparent bottom.

Programming tips

Make sure you plug in the sensors that your program needs! If you use the WAIT FOR CLAP block, you will need the "ear" (sound) sensor. If you use the LIGHT or DARK parameter cards, you will need the "eye" (light) sensor. If you use the NEAR or FAR parameter cards, you will need the "telescope" (distance sensor). If you use the RED/WHITE/BLUE LIGHT ON blocks, you will need the light bulb.

Take care of your motor modules!

KIBO's motor modules are designed to turn KIBO's wheels; they are not designed to carry a lot of weight. So, please don't force KIBO to go faster than it wants, and don't push down on KIBO's body when its wheels and motors are installed. These behaviors can damage the motor modules. Our warrantee doesn't cover damage caused by improper motor use.

Scanning tips

To scan, hold KIBO 2–4" away from the bar code. Shine the red scanner light onto the bar code. It's ok if the light is a little "bigger" than the bar code.

If KIBO won't scan, try changing KIBO's position slightly. Move it slightly closer or farther away from the block and try changing the angle a little bit.

This type of reflection can interfere with scanning.

If you're still having trouble scanning, notice if there is light reflecting from your stickers. Try moving away from direct overhead lighting and windows. Or, try scanning the sides of the blocks, instead of the top.

If scanning from the top isn't working ...

Try scanning from the side!

Join our KIBO community!
Sign up for our email newsletter for KIBO news, activity ideas, classroom tips, and more, at
http://kinderlabrobotics.com

Visit us on Facebook **http://facebook.com/KinderLabRobotics**

Follow us on Twitter: **@KinderLabRobot**

Colors of some components may vary.
Additional parts available at **shop.kinderlabrobotics.com**
Complete parts list at **kinderlabrobotics.com/compare**

References

Bers, M. U. (2008). *Blocks to robots: Learning with technology in the early childhood classroom*. New York, NY: Teachers College.

Bers, M. U. (2012). *Designing digital experiences for positive youth development: From playpen to playground*. Cary, NC: Oxford.

Bers, M. U. (2018). *Coding as a Playground: Programming and Computational Thinking in the Early Childhood Classroom*. New York, NY: Routledge Press.

Darragh, J. C. (2006). *The environment as the third teacher*.
Retrieved from http://www.eric.ed.gov/PDFS/ED493517.pdf

International Society for Technology in Education (2017). *ISTE Standards for Students*.
Retrieved from https://www.iste.org/standards/for-students

K–12 Computer Science Framework. (2016). Retrieved from http://www.k12cs.org.

Massachusetts Department of Education (2016). *Digital Literacy and Computer Science Framework*. Retrieved from https://www.doe.mass.edu/frameworks/dlcs.pdf

National Governors Association Center for Best Practices & Council of Chief State School Officers

(2010). *About the Standards*. Retrieved from http://www.corestandards.org/about-the-standards/

Papert, S. (1993). *Mindstorms: Children, computers, and powerful ideas*. New York, NY: Basic Books.

Virginia Department of Education (2017). *Computer Science Standards of Learning* (SOL). Retrieved from http://www.doe.virginia.gov/testing/sol/standards_docs/computer-science/index.shtml

Vygotsky, L. S. (2012). *Thought and Language*. Cambridge, MA: The MIT Press.