EXHIBIT 11 – Mandatory Technical Specifications

4.1 Mandatory Requirements - Curriculum

4.1.1 Curricula must be aligned to the CSTA Standards.

Yes. CoderZ is Aligned to CSTA Standards. Please see ***Exhibit 13 – Curriculum*** for a complete overview of CoderZ alignment to standards.

4.1.2 Must prepare teachers to teach the provided computer science curriculum by the start of the 2021-22 school year.

Yes. CoderZ offers ongoing professional development for teachers that introduces them not only to CoderZ, but to the instructional theory and practices for high quality STEM, coding, and virtual robotics instruction. *Please see **Exhibit 14 - Professional Development**,* which includes our approach and timeframe to Preparing Teachers to Teach Computer Science.

4.1.3 Curriculum grade level(s) must be identified

Yes. CoderZ provides grade level recommendations to guide instructional practice and delivery for each course.

4.1.4 Length of curriculum (unit, semester, full year) and model of delivery (traditional classroom, virtual, or blended) must be provided. After school curriculums are not considered this round.

Yes. CoderZ includes comprehensive teacher guides, lesson plans, and additional resources which provide multiple instructional delivery recommendations including, but not limited to, traditional classroom, virtual, or blended. CoderZ also provides recommended timeframe(s) for instructional delivery.

4.1.5 Curriculum must be designed to be offered by classroom teacher.

Yes. CoderZ provides detailed teacher guides and a wealth of tools to help teachers introduce STEM concepts into their classroom(s) and classroom teacher guidance optimizes instruction with CoderZ.

4.2 Mandatory Requirements – Professional Development

4.2.1 Must include professional development that is delivered by the Respondent around implementation of the curriculum

Yes. CoderZ provides both introductory and comprehensive professional development. Please refer to ***Exhibit 14 – Professional Development***, which includes the CoderZ approach to and delivery options for Professional Development.

4.2.2 Professional development must be available before the beginning of the 2021-22 school year.

Yes. CoderZ offers professional development for teachers that introduces them not only to CoderZ, but to the instructional theory and practices for high quality STEM, coding, and virtual robotics instruction and will be available before the beginning of the

2021-22 school year. *Please see **Exhibit 14 - Professional Development**,* which includes our approach and timeframe to Preparing Teachers to Teach Computer Science.

4.2.3 Information about whether aligned professional development is required in order to use the provided curriculum

Yes. CoderZ offers, *but does not require*, professional development for teachers. CoderZ includes comprehensive teacher guides, lesson plans, and additional resources which provide multiple instructional delivery recommendations including, but not limited to, traditional classroom, virtual, or blended.

4.3 Implementation

Upon award of a Contract for services the Agency shall negotiate an implementation schedule with the successful Respondent.

Yes. CoderZ has deep experience implementing its platform in educational settings around the country and will work specifically with the Agency and all districts to map out a successful implementation schedule, including key deliverables, milestones, and outcomes.

EXHIBIT 12 –Program Overview

Describe the origin of the program.

Guided by its vision to make CSTEM more inclusive, CoderZ – a web-based platform through which students of all ages can learn how to code real and virtual robots – was launched by Intelitek, a world leader in STEM solutions for over 35 years. Our programs have been taught in in over 50 countries, educating students in more than 40,000 schools. Educational institutions have implemented our programs in various ways, including:

- Blended Learning: Combining theoretical learning with hands on labs
- Remote Learning: Delivering content and simulation software exercises.
- Hybrid Learning: Maintaining continuity of learning as students navigate new learning environments.

We understand the changing needs of technology classrooms and have designed flexible solutions that meet those needs even as they change rapidly over time.

Provide the length of time the program has been offered

CoderZ was launched by Intelitek, Inc. in 2017 for school and district implementations.

Provide the content covered and a description of the curriculum.

CoderZ is a powerful, easy to use, award-winning online platform through which students learn valuable STEM skills such as coding, robotics and physical computing. CoderZ is designed for use in the classroom or through a wide range of remote learning environments.

- Students experience the platform as a fun, gamified environment thanks to easy-to-use code editors and 3D virtual robot simulations. As students run their code, they get immediate feedback from the simulation and remain highly motivated to accomplish their missions.
- Students learn coding skills by completing missions with their virtual robot. Each mission focuses on a specific skill. The students are supported with curriculum, tips & hints, a knowledge center that contains sample code – as well as through the feedback received by the simulation. The game like environment of learning ensures students are engaged and motivated to learn.
- Teachers see a timely, relevant, and highly engaging platform that aligns to standards, provides in-depth support, and delivers immediate feedback enabling personalized instruction.
- Our embedded professional development resources include a range of ever-ready materials for teachers, including teacher guides, pacing guides, coding solutions, discussion points and assessments. By integrating teacher support resources directly into the application, teachers with no coding/robotics background are empowered to succeed with CoderZ .
- Building on this theme of teacher support, the platform offers multiple types of scaffolding for teachers including, curriculum; student learning objectives; teacher guides, pacing guides, instructional guides, mission solutions, assessments, and professional development.
- The platform caters to both novice and expert student users, offering a simple, block based visual editor for new coders or allowing more advanced students to start working directly with Text based Java or Python code editing.

- CoderZ offers different programs to elementary, middle and high school students, as well as a virtual coding competition that generates unparalleled excitement and enthusiasm for STEM.
- CoderZ is cloud based, accessible online from the classroom or at home and puts the world of coding and robotics in the hands of every student. All that is required to get started is connectivity and a simple Chrome book.
- The CoderZ curriculum is aligned to Next Generation Science Standards (NGSS), and K-12 Computer Science Standards (CSTA – Computer Science Teachers Association).

Curriculum Currently Available

CoderZ Adventure: CoderZ Adventure introduces your students to the fascinating world of CSTEM and robotics through an exciting journey and adventure in CoderZ world. Students will learn how to program a virtual robot to navigate in CoderZ Frozen Island, the Lost City, Candy Town, and other exciting locations while applying basic math, geometry, and more.

- Designed for grades 3-5
- Approximately 15-20 hours of instruction

Cyber Robotics 101: Cyber Robotics 101 is a flexible learning program for educators to introduce students to the core concepts of code development and robotics. Students will learn mechanics, navigation, sensors and more while being introduced to programming components like commands, variables, conditional logic, loops, smart blocks (functions) and more.

- Designed for grades 5-8

- Approximately 25 hours of instruction

Cyber Robotics 102: Cyber Robotics 102 builds on the skills developed in CR-101, only this time in a physical environment. Students are introduced to concepts such as acceleration, gravity, torque. In addition, students are introduced to advanced application of sensors and PID algorithms.

- Designed for grades 7-10
- Approximately 25 hours of instruction

Python Gym: Python Gym introduces students to the exciting world of programming using the Python language. Students will gain a basic understanding of object-oriented programming and enhance their critical thinking and problem-solving skills as they learn to design, code, and debug Python programs. Challenging missions encourage them to master important programming concepts such as variables, operators, and control flow constructs.

- Designed for grades 8-12
- Approximately 25 hours of instruction

Amazon Cyber Robotics Challenge: Amazon Cyber Robotics Challenge introduces students to the exciting world of programming in an Amazon Fulfillment Center. Students are guided through novice level coding and robotics skills and concepts by two Amazon Future Engineer interns. Students are introduced to how robots and associates work together at an Amazon Fulfillment Center to process orders safely and efficiently.

- Designed for grades 3-12
- Approximately 3 hours of instruction

Describe the professional development delivered by the Respondent around implementation of the curriculum

CoderZ is an online educational environment that fosters 21st century skills while having fun as they program their own virtual robot. What makes CoderZ so special is that the product and supporting services – including our professional development offerings – help teachers learn and grow in much the same way we help students do the same.

With that said, an education platform is only as powerful as the people behind it, which is why we have developed a professional development roadmap that ensures teachers are equipped to deliver high quality STEM, coding, and robotics instruction. We emphasize core teaching and learning principles that, while anchored to the CoderZ experience, are transferable to all aspects of STEM instruction.

Teacher Resources

Every CoderZ teacher has access to a portfolio of resources specifically designed to set them up for success. Even with no computer science background, teachers can engage in CoderZ materials and professional development quickly and confidently. Our Teacher Resources portfolio continues to expand, and already includes:

- Student outcome plans
- Lesson plans tied to CoderZ modules
- Speaking notes
- Pacing guides
- Questioning techniques, including specific questions for use in the classroom
- Discussion and writing prompts
- Assessment guides, including how to use progress monitoring for formative assessment

Teacher Professional Development Course

Teachers in districts that adopt CoderZ will have immediate access to our standard Professional Development course, which includes 12 – 20 hours of training that sets them on the path to teaching computer science curriculum well within the 6 – 12-month time frame noted in this RFP.

Our course work includes:

- An introduction to Computer Science principles and the world of educational robotics
- Pedagogy of discovery-based learning in Computer Science
- Teaching strategies for blended learning
- Integrating math and science into problem solving in Computer Science
- Leveraging student engagement data and reporting tools to apply personalized teaching interventions
- Strategies for teaching in a gamified environment that maintain validity and reliability
- Incorporating key 21st-Century skills into Computer Science instruction, including:
    - Collaboration
    - Effective listening
    - Engaging in discussions
    - Giving and receiving constructive feedback further develop ideas
- An introduction and overview of the CoderZ online curriculum, including how the curriculum maps to CSTA standards, how it can be applied in traditional classrooms, hybrid learning scenarios, or virtual learning environments.
- Simulation and practice with various teaching environments and classroom scenarios.

CoderZ professional development can be delivered on-site or online, though we anticipate utilizing fully online delivery of our professional development courses in Iowa at least through the 2021-2022 school year to ensure safety and to reduce expense. We are mindful of the precautions required by COVID-19 and view online delivery as a cost-saving maneuver for districts without sacrificing quality.

By the time teachers have completed our coursework and brought students through the first missions of CoderZ, they leave with a deep understanding of computer science fundamentals, even if they bring no experience into the work at the outset. Further, teachers are able to immediately apply fundamental concepts in the classroom (in-person or online) with the support of our instructional guides, teacher resources, formative assessments, and reporting tools. Below are samples of Professional Development resources that relate to the curriculum samples added to Exhibit 12:

CoderZ

The Touch Sensor

Think of 4 different touch sensors that you use in your everyday life.

Consider where they are located.

Journal about each of the touch sensors that you thought of:

What is its purpose?

If you would be the designer of it, how would you design it to best fit your needs?

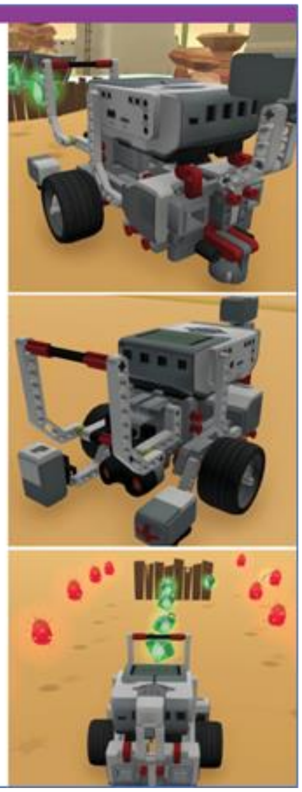
Why in your opinion the touch sensor in metter, was designed as it is now?

Discuss what you just journaled with your pair.

In groups of 4, discuss the following:

- What material goals did this discussion cover?
- What pedagogical goals, did this activity cover?
- How does these questions, serve the learning of the touch sensor?

Come up with 3 different questions, that can have the same learning outcomes, and pedagogical outcomes we talk about.



CoderZ

CoderZ Philosophy

CoderZ Curriculum Values

Our core values

Basics of Computer Science is a must for all students. We believe that fundamentals of Computer Science and fundamentals of Robotics are a must in our fast-paced technological world. For all people. Understanding of the importance of exact communication, of checking oneself, of taking full responsibility for learning from mistakes, feeling the value of the 'why' question, stands in the basis of becoming an active contributor to society. We believe that self esteem comes from feeling control in one's life. This can be achieved, by growing and establishing the feeling of being able to find solutions for problems. We find Computer science and robotics as great tools to gain all theses, as well as important skills to learn.

Student as Active learners

We believe that students learn best when they are active learners. Our gamified platforms are designed to make all learning active and engaging, assisting teachers in motivating students. We encourage students to try out different strategies when solving problems, while gaining more tools and knowledge in CSTEM.

Teachers as Social change generators

We believe every change we want to make for our students, must begin empowering their teachers. Providing the tools and strategies to teachers. We train teachers to use different strategies in class for empowering their students, and we give teachers tools, that can be used in different ways, so they can adjust to their diverse and specific students they teach.

In addition, the teacher guide and the student learning outcomes provide significant guidance to the teacher on each and every lesson:

Objectives

Guiding Question: How can planning help to make better programs?

Learning Objectives and Assessment:

Describe the benefits of planning an algorithm before beginning to code

- In the Reflection Questions, ensure that students understand that planning can help them to understand what the code is supposed to do before coding, saving time and frustration that can come from a "guess and check" method.

Decompose a problem into its components

- Check student success on Mission 7, which requires that they break down the robot path into its component parts.

Students will:

Modify drive blocks to accept a distance parameter

Use measurements to assist with planning a solution

Standards

- CSTA 1B-AP-11 Decompose (break down) problems into smaller, manageable subproblems to facilitate the program development process.

The teacher guide provides explanations and instructional resources relevant to each lesson and the activities assigned therein. The text sample below demonstrates how our professional development methodology strives for relevance in the classroom not only in its subject matter, but in its tone and delivery. It's critical that we get into the hearing of teachers so that they can enter into the work with great confidence and enthusiasm.

(BEGIN SAMPLE)

Mission 6
The mission is significantly more difficult from the ones the students have seen so far for two reasons: it involves more steps and students are asked to write all their code from scratch, rather than just arranging given blocks. The mission also introduces a new game element, moving platforms controlled by buttons. You should have your class discuss and work on the first part of this mission together, before allowing students to complete it on their own.

Let students watch the tutorial for this mission. It explains that the button needs to be pressed to create a bridge to reach the target but does not give explicit instruction for solving the mission. Remind students of the discussion from the introduction about breaking down a problem into subproblems.

Let students **think** and journal for 60sec: How can the mission be broken down into two smaller problems?

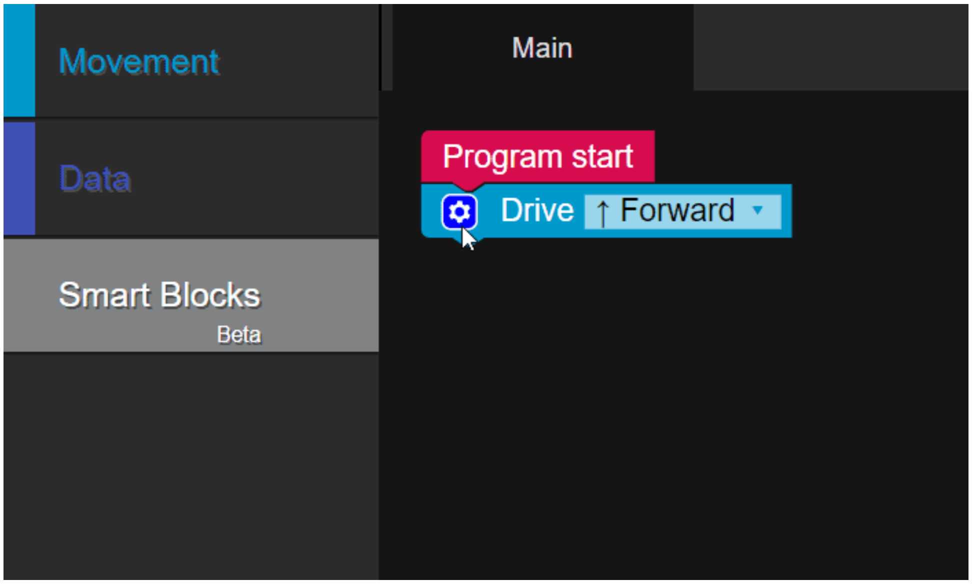
Now let the students **pair** with one of their peers (split into pairs either using the virtual session's capabilities, or in the class) (Each student gets 60 seconds to explain her Idea, and the other - listens. They switch roles after 60 sec)

Share the thoughts. Let each student share their idea.
Something like this might come up:
- Get the robot from the starting point to the button
- Get the robot from the button to the target

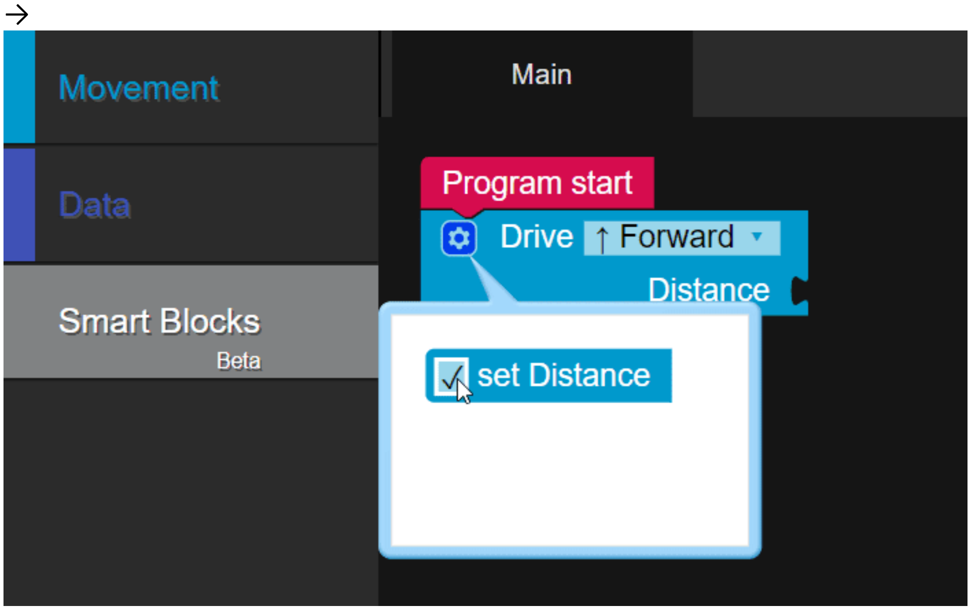
Now ask students to break that first step down further. What does the robot need to do to get from the starting position to the button? Pretty clearly it needs to drive a distance, turn, and drive a little more. But how far to drive each time.

Have students switch to Explore Mode. Not only does this top-down view let them see the entire environment, but it also reveals some annotations that have been added to show the important distances. From this they should see that the first step the robot needs to take is to drive forward 8 meters.

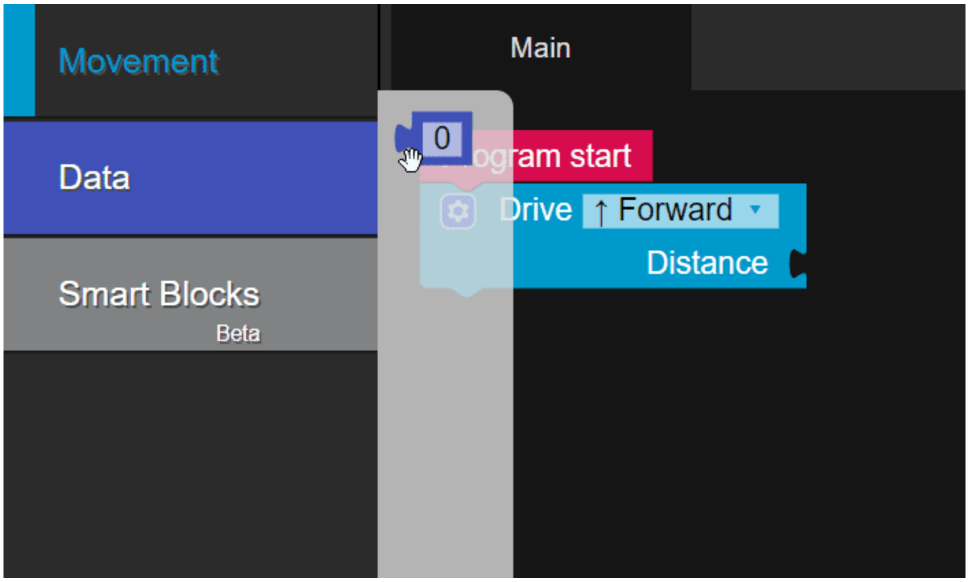
Since the default Drive block available in the menu is not ready for a distance parameter, students may need a reminder of how to add one. Click on the  button, check "set Distance", and drag and attach a number block from the Data menu.


Movement
Data
Smart Blocks
Beta
Main
Program start
Drive
↑ Forward

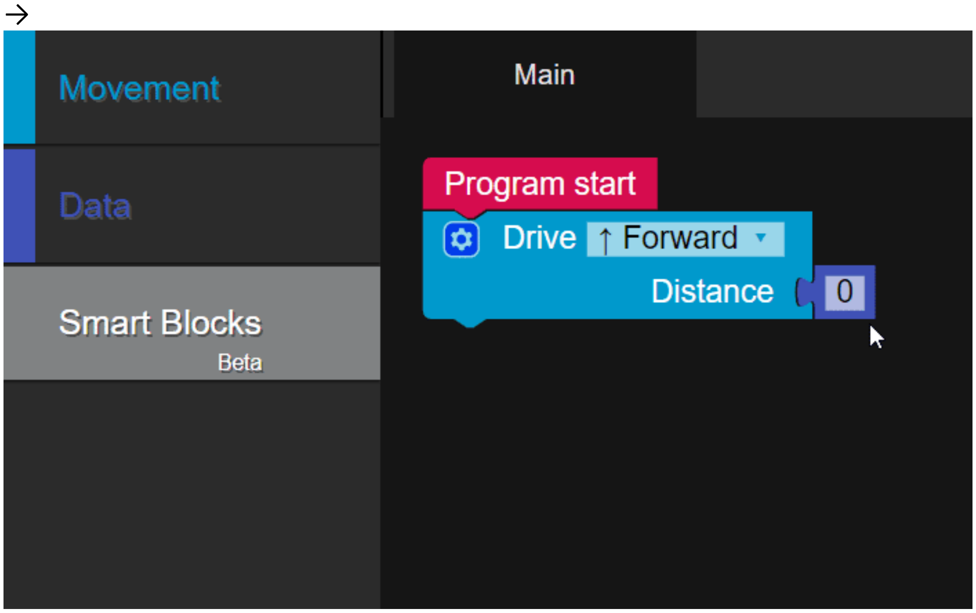
→


Movement
Data
Smart Blocks
Beta
Main
Program start
Drive
↑ Forward
Distance
set Distance

→



→



At this point students should be in a position to write the three steps necessary to get the robot to the button (Drive 8; Turn Left; Drive 2). Let them do so and then test their solutions by running the simulation. Once they have successfully navigated the robot to the button, they can then work on getting the robot from the button to the target. Remind them that they need to press the Reset Simulation button between tests, so the robot returns to its starting position.

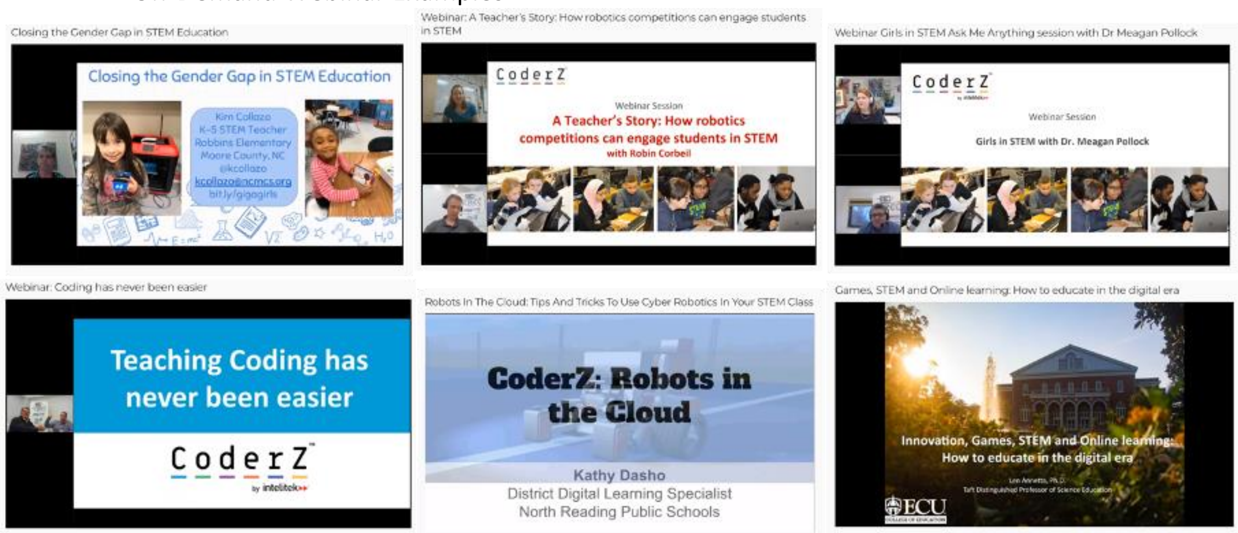
If anyone needs help with the second part of the program you can point out that when the robot reaches the button the target is behind it. That means the robot will either need to turn around or move backwards to teach the target. Students should review missions 4 and 5 if they need a refresher on turning around and moving backwards.

One other tip that will be helpful to everyone: If you duplicate (using the right-click menu) a block that is already configured with a distance parameter, the new block will have the parameter as well. That can be an easier way to add multiple Drive distance blocks than building each one separately.

(END SAMPLE)

In addition to a growing library of teacher resources and our professional development courses, we deliver on-demand webinars on a regular basis. We recruit from within our ranks as well as from a global network of expert educators and technologists, each of whom bring timely, relevant training directly to educators.

On-Demand Webinar Examples



Ensure the program can be offered during the school day rather than after-school.

Curriculum Delivery
Each lesson in the CoderZ curriculum begins with a definition of the **student learning outcomes,** clearly indicating the knowledge, skills and abilities individual students should possess and demonstrate upon completion of a learning experience or sequence of learning experiences. With that as context, the CoderZ teaching and learning experience comes with comprehensive support for both teachers and students.

For Teachers:

- The **teacher guide** lists out the **resources** available for each lesson. These resources will include videos, articles, curriculum materials, and gamified missions. as well as suggestions for assessment opportunities and techniques, reflection exercises, and discussion topics.
- The **pacing guidelines** provides guidelines as to how much time should be allocated for each activity in the session.
- The presentations supplement the theoretical background delivered in each lesson. The presentations may be used by an instructor to introduce a certain subject or for self - paced learning by a student. The presentations contain sample code and examples of

problem resolution. Following the presentation, students are directed towards 'Play Time'.

For Students:

- During play time – students attempt to solve gamified challenges by coding the virtual robot to perform tasks that will bring to the resolution of a problem.
- The curriculum directs students to **focus on solutions**. Through creative and analytical thinking and with the help of peer discussion, students aim to find the best possible solutions to the challenges to solve problems. As these problems become more complex, students will develop a more systematic approach to problem solving involving planning, data collection, analysis, application and evaluation. Many Certain activities promote will require students to work collaboratively in order to find reach the required solution
- CSTEM is interdisciplinary in its nature – and so are the challenges faced by the students. This means that students will need to apply data collection, math, concepts in physics, understanding in robotics and code – in order to solve challenges.
- The curriculum provides opportunities for learners to interact with the material in a variety of ways. Complex concepts are modelled in the physical world, allowing students to engage with them in a more approachable way before using them to program. These understandings are then reinforced with class discussion and reflection. The activities are designed to work in both an in-classroom or virtual format.
- Following play time, the instructor is provided with tools to encourage reflection on learning through discussion or through evaluation. Although the instructor has visibility as to the progress of the students – the reflection is designed to reinforce that the learning outcomes have been achieved.

Stand Alone / Integrated Curriculum

The CoderZ learning center provides students with access to all the required resources including presentations, videos, gamified challenges, projects, assessments and more. The student remains in a secure cloud-based environment and migrates seamlessly from one activity to the next.

The benefit of our approach is that CoderZ can operate as a fully independent teaching and learning experience, serving students of all skills and backgrounds by leveling the different modules accordingly. With that, teachers can also integrate into existing and future curriculum for several reasons:

- CoderZ utilizes CSTA standards to drive its pedagogical approach and ensure alignment to STEM teaching priorities
- CoderZ incorporates programming languages that are recognized as critically important across all industries, including Java and Python.
- CoderZ integrates math and science together, allowing teachers to weave missions into their existing scope and sequence.

With CoderZ, students across Iowan will learn how to code virtual robots accompanied by a step-by-step curriculum with gamified missions, all the while mastering skills deemed critical by CSTA standards.

Describe how the curriculum and professional development can fit into a K-12 CS plan, as required in HF 2629.

House File 2629, signed into law in 2020, built on Senate File 274 by requiring computer science instruction. High schools must offer at least one high-quality one-semester course starting July 1, 2022. Middle schools must provide high-quality computer science in seventh or eighth grade by July 1, 2023. Elementary schools must provide high-quality computer science in at least one grade level by July 1, 2023. Schools and the state also must develop K-12 computer science plans by July 1, 2022.

CoderZ developed all courses as age-appropriate points of entry into computer science that sets students up for success, no matter their background or experience. All courses are leveled for experience and may be implemented at elementary, middle, and high school with an average instructional time of 20-25 hours per course perfectly supporting House File 2629 at all levels.

With nearly 100 hours of curriculum and additional extra-curricular activities, such as CoderZ League – the nation's leading cyber robotics competition - CoderZ is both comprehensive and flexible enough to fit into a K-12 plan. They key element to working with CoderZ is that our platform and our people are highly adaptable and willing to work with schools to suite their specific needs. Adoption varies from after school enrichment activities to full scale CSTEM program; some of the most frequent scenarios including:

- **District Adoption**
When planning large scale implementation of a robotics program, CoderZ is a perfect fit. Districts can use the various curriculums offered by CoderZ to different grade levels in the district. By using the CoderZ reporting tools, the district administration can get a clear picture of progression in the district.

- **Exploring STEM**
CoderZ is adopted by schools as one of several programs that are offered to students that will explore several STEM areas during the academic school year. These topics may include: Solar Energy; Electric Vehicles; Video editing and Robotics. In such cases, CoderZ will be used as the curriculum used for exploring robotics.

- **Scaling Robotics Programs**
Schools with limited budgets for purchasing physical robots will adopt CoderZ and by that enable more students to take robotics. Students can now work in groups, when some of the students carry out projects on the physical robots, while others will focus on developing their computational skills, coding a virtual robot.

- **STEM Electives**
Certain schools offer different STEM topics as full semester or full year electives. CoderZ supports a progression from introduction to basic concepts in robotics and coding, through to advanced programing of complex algorithms.

- **After School**
CoderZ is adopted by after school clubs that offer enrichment activities to students in coding and robotics. The activity could be completely virtual or tied to a physical robotics activity. Many of the after-school clubs will also participate in the CoderZ League (competition).

Implementation Overview
Standing behind our world-class platform are the people dedicated to supporting educators every day. We work with schools every step of the way, from onboarding to training to support, to ensure that you move through the implementation phase as quickly as possible and drive towards full adoption of the platform. It is our goal to see CoderZ become part of the fabric of every school's learning community, so we organize the CoderZ rollout into key phases (customized for every engagement):

- Onboarding
    - Identifying key stakeholders in the rollout process and organizing our game plan, including timelines, milestones, and key deliverables
    - Identifying key technical, training, and communication needs required to move forward

- Implementation
    - Platform activation and account set up
    - Whitelisting web links with IT administration
    - Utilization of Single-Sign-On and rostering platforms, as needed, such as ClassLink, clever, and Google
    - Review of key learning standards and learning objectives to ensure teachers are using a common language and that our shared stakeholders are working in alignment

- Ongoing Training and Support
    - Onboarding webinars customized to your district
    - Designated education success manager focused on delivering training and support to all stakeholders in the district
    - Knowledge hub for self-service support and self-paced training

- Adoption
    - Level-setting expectations for each quarter, semester, and the full-year to ensure we are working towards specific utilization goals
    - Utilization reporting and check-ins, including support and interventions as needed for classrooms that have yet to meet goals

With every school and district engagement, we strive for full adoption. While some platform providers are satisfied with introducing and activating technologies in schools, we are committed to seeing every district can benefit from true adoption of CoderZ.

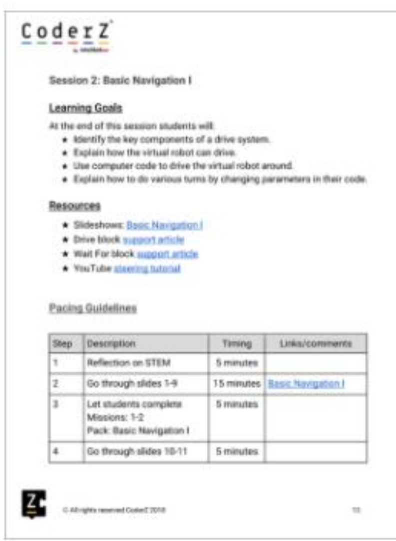
There are no third-party respondents involved in our implementation strategy. Our in-house team works directly with our school customers to ensure strong onboarding, implementation, and adoption. This holds true, as well, for our ongoing support of the platform.

There is no STEM, coding, or robotics expertise or skill set required by personnel from the State of Iowa. CoderZ is designed to be accessible and available to all skill levels, so personnel need only bring their perspective on best-practices for a successful for a rollout of the platform at a state, district, or school level.

In terms of time requirements, we anticipate initiating a series of virtual meetings with representatives from the state, lasting approximately 60 minutes each, so that our team understands the nuanced requirements of schools and districts in Iowa and to ensure that our work aligns with initiatives that the State is implementing in parallel. We recognize that the State will have many important initiatives running at any given time, as will schools and districts, so we want to make sure that our efforts to foster CoderZ adoption work in concert with the priorities and plans of the State.

In terms of documentation that is provided with the CoderZ platform, we include a robust portfolio of resources and documentation that support the onboarding, implementation, and full adoption of the platform, all at *no additional cost.*

- Every course includes a Teacher's Guide integrated directly into the platform, providing fast, easy access to a critical resource



CoderZ

Session 2: Basic Navigation I

Learning Goals

At the end of this session students will:

- Identify the key components of a drive system.
- Explain how the virtual robot can drive.
- Use computer code to drive the virtual robot around.
- Explain how to do various turns by changing parameters in their code.

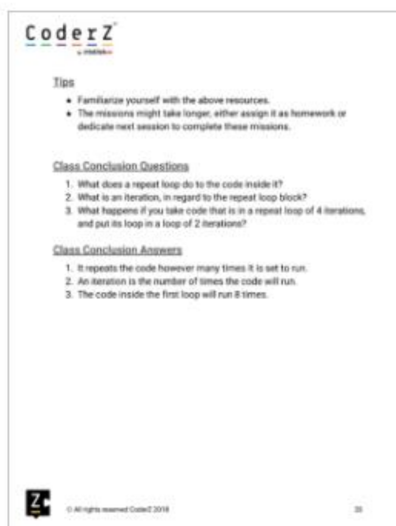
Resources

- Slideshows: Basic Navigation I
- Drive block support article
- Wait For block support article
- YouTube steering tutorial

Pacing Guidelines

Step	Description	Timing	Links/comments
1	Reflection on STEM	5 minutes	
2	Go through slides 1-9	15 minutes	Basic Navigation I
3	Let students complete Missions: 1-2 Pack: Basic Navigation I	5 minutes	
4	Go through slides 10-11	5 minutes	

© All rights reserved CoderZ 2018



CoderZ

Tips

- Familiarize yourself with the above resources.
- The missions might take longer, either assign it as homework or dedicate next session to complete these missions.

Class Conclusion Questions

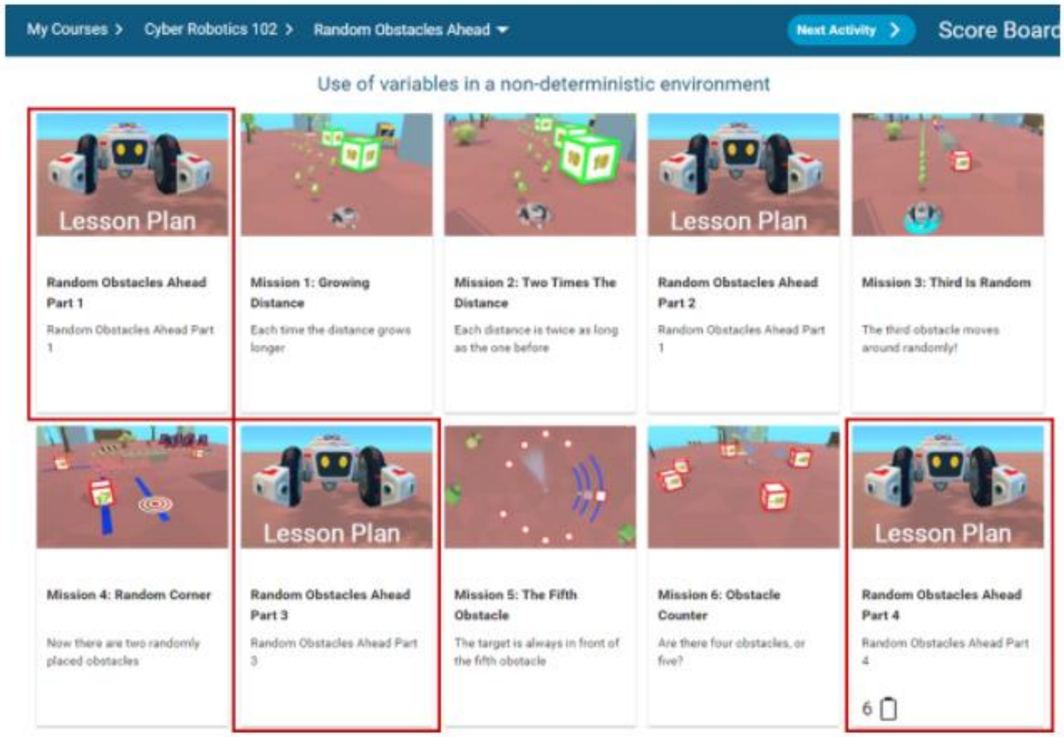
1. What does a repeat loop do to the code inside it?
2. What is an iteration, in regard to the repeat loop block?
3. What happens if you take code that is in a repeat loop of 4 iterations, and put its loop in a loop of 2 iterations?

Class Conclusion Answers

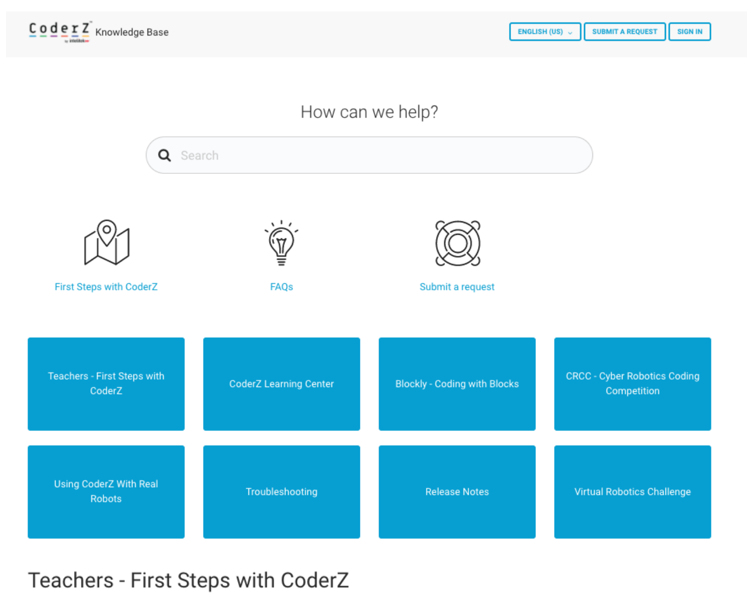
1. It repeats the code however many times it is set to run.
2. An iteration is the number of times the code will run.
3. The code inside the first loop will run 8 times.

© All rights reserved CoderZ 2018

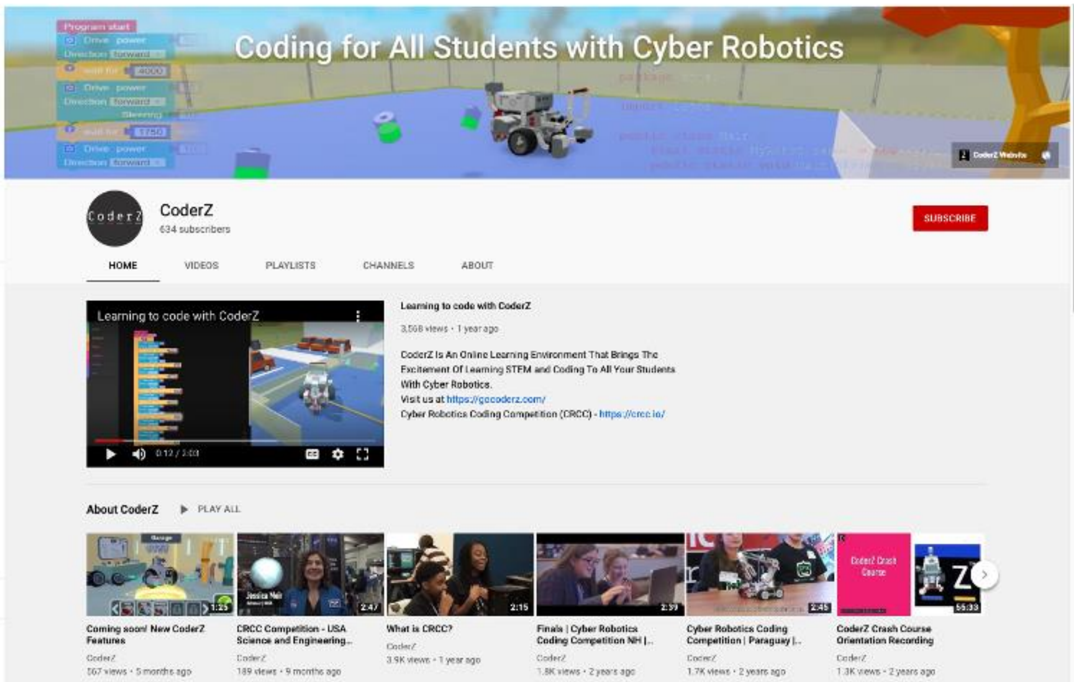
- Teachers have access to ready-made Lesson Plans that help guide instruction and ensure alignment to standards



- Our Knowledge Base is available to all users, offering an in-depth "how-to" that helps teachers and master the platform quickly, ensuring quick implementation and sustained support for all



- We host a popular YouTube channel with ongoing updates and presentations by the CoderZ team as well as by educators who share insights and advice on how to make the most of the platform.



Describe how the curriculum engages diverse learners.

CoderZ is guided by its vision to make CSTEM more inclusive:

- Web-based platform through which students of all ages can learn how to code real and virtual robots.
- Success at all levels, no matter what previous experience a student or teacher has with coding and robotics. Course levels include novice, beginner, intermediate and advanced to ensure all students have an opportunity to be successful.
- Virtual instruction with multiple implementation models provides the maximum opportunity for engaging diversity.
- 3-D high level graphics in a game centered outcome based instructional approach also encourages and engages all learners including under-represented populations.
- CoderZ wants to find the students who didn't know they loved coding and robotics.

Describe how the curriculum connects to the world of work and proof of success.

CoderZ courses bring math and science to life through 3-D engaging games to support understanding of multiple levels of potentially complex concepts. Specifically, CoderZ course, Amazon Cyber Robotics Challenge, prepares students for the real world of how coding and robotics positively impact the world of work.

Amazon Cyber Robotics Challenge introduces students to the exciting world of programming in an Amazon Fulfillment Center. Students are guided through novice level coding and robotics skills and concepts by two Amazon Future Engineer interns. Students are introduced to how robots and associates work together at an Amazon Fulfillment Center to process orders safely and efficiently.

Each of the four leveled CoderZ courses, CoderZ Adventure, Cyber Robotics 101, Cyber Robotics 102, and Python Gym teach coding and robotics skills and concepts that are transferrable as students work with four different robots and scaffolded instruction builds knowledge through depth of content.

Additionally, My Projects provides a sandbox for teachers and students to connect to the world of work through outcome based instructional models where teachers and students determine the goals and how to accomplish those goals with the robot through coding. Students may use Blockly or text-based code in My Projects providing flexibility to try different algorithms in a variety of virtual scenes.

Exhibit 13 – Curriculum

Curriculum Delivery

Each lesson in the CoderZ curriculum begins with a definition of the **student learning outcomes,** clearly indicating the knowledge, skills and abilities individual students should possess and demonstrate upon completion of a learning experience or sequence of learning experiences. With that as context, the CoderZ teaching and learning experience comes with comprehensive support for both teachers and students.

For Teachers:

- The **teacher guide** lists out the **resources** available for each lesson. These resources will include videos, articles, curriculum materials, and, gamified missions. as well as suggestions for assessment opportunities and techniques, reflection exercises, and discussion topics.
- The **pacing guidelines** provide guidelines as to how much time should be allocated for each activity in the session.
- The presentations supplement / compliment / outline the theoretical background to be learned and/or to be delivered in each lesson. The presentations may be used by an instructor to introduce a certain subject or for self-paced learning by a student. The presentations contain sample code and examples of problem resolution. Following the presentation, students are directed towards 'Play Time'.

For Students:

- During play time – students attempt to solve gamified challenges by coding the virtual robot to perform tasks that will bring to the resolution of a problem.
- The curriculum directs students to **focus on solutions**. Through creative and analytical thinking and with the help of peer discussion, students aim to find the best possible solutions to the challenges needed to solve problems. As these problems become more complex, students will develop a more systematic approach to problem solving involving planning, data collection, analysis, application and evaluation. Many Certain activities promote will require students to work collaboratively in order to find reach the required solution
- CSTEM is interdisciplinary in its nature – and so are the challenges faced by the students. This means that students will need to apply data collection, math, concepts in physics, understanding in robotics and code – in order to solve challenges.
- The curriculum provides opportunities for learners to interact with the material in a variety of ways. Complex concepts are modelled in the physical world, allowing students to engage with them in a more approachable way before using them to program. These understandings are then reinforced with class discussion and reflection. The activities are designed to work in both an in-classroom or virtual format.
- Following play time, the instructor is provided with tools to encourage reflection on learning through discussion or through evaluation. Although the instructor has visibility as to the progress of the students – the reflection is designed to reinforce that the learning outcomes have been achieved.

Stand Alone / Integrated Curriculum

The CoderZ learning center provides students with access to all the required resources including presentations, videos, gamified challenges, projects, assessments and more. The student remains in one cloud-based environment and migrates seamlessly from one activity to the next.

The benefit of our approach is that CoderZ can operate as a fully independent teaching and learning experience, serving students of all skills and backgrounds by leveling the different modules accordingly. With that, teachers can also integrate into existing and future curriculum for several reasons:

- CoderZ utilizes CSTA standards to drive its pedagogical approach and ensure alignment to STEM teaching priorities
- CoderZ incorporates programming languages that are recognized as critically important across all industries, including Java and Python.
- CoderZ integrates math and science together, allowing teachers to weave missions into their existing scope and sequence.

With CoderZ, students across Iowan will learn how to code virtual robots accompanied by a step-by-step curriculum with gamified missions, all the while mastering skills deemed critical by CSTA standards.

Curriculum Currently Available:

- **CoderZ Adventure**: CoderZ Adventure introduces your students to the fascinating world of CSTEM and robotics through an exciting journey and adventure in CoderZ world. Students will learn how to program a virtual robot to navigate in CoderZ Frozen Island, the Lost City, Candy Town, and other exciting locations while applying basic math, geometry, and more.
    - Designed for grades 3-5
    - Approximately 15-20 hours of instruction

- **Cyber Robotics 101:** Cyber Robotics 101 is a flexible learning program for educators to introduce students to the core concepts of code development and robotics. Students will learn mechanics, navigation, sensors and more while being introduced to programming components like commands, variables, conditional logic, loops, smart blocks (functions) and more.
    - Designed for grades 5-8
    - Approximately 25 hours of instruction

- **Cyber Robotics 102:** Cyber Robotics 102 builds on the skills developed in CR-101, only this time in a physical environment. Students are introduced to concepts such as acceleration, gravity, torque. In addition, students are introduced to advanced application of sensors and PID algorithms.
    - Designed for grades 7-10
    - Approximately 25 hours of instruction

- **Python Gym:** Python Gym introduces students to the exciting world of programming using the Python language. Students will gain a basic understanding of object-oriented programming and enhance their critical thinking and problem-solving skills as they learn to design, code, and debug Python programs. Challenging missions encourage them to master important programming concepts such as variables, operators, and control flow constructs.
    - Designed for grades 8-12
    - Approximately 25 hours of instruction

Curriculum Alignment to Iowa/CSTA:

Each module of the curriculum includes a framework for the expected student learning outcomes, aligned to both CSTA and NGSS Engineering Design standards. This framework provides the context for all learning and assessment in the course. Lessons within the modules include specific learning objectives and associated assessment opportunities that reference the overall framework, as well as a list of CSTA standards addressed within the lesson itself.

The alignment contains the identifier of the standard in question, the relevant grade level, and the concept being addressed. The CSTA standard is a guideline for STEM Educators nationwide and to this end, we have aligned the CoderZ curriculum to map to the 7 practices laid out in the standards.

- **Communicating about Computing**
  Lessons cover comparing various algorithms and predicting their outcomes, and justifying which hardware and software is used for specific tasks. CoderZ also enables students to explain how computer science makes a positive change to the workforce and how to use computational skills to analyze data and solve problems.

- **Collaborating**
  As well as working in various team formats, CoderZ supports students in soliciting and integrating peer feedback to improve on their own learning and development.

- **Creating Computational Artifacts**
  Both in teams and independently, CoderZ supports students in developing programs that include complex sequences. They also create variables that represent different types of data and practice manipulating these.

- **Developing and Using Abstractions**
  Being able to use and modify procedures which have been created elsewhere for other tasks, understanding how they can be reapplied. CoderZ enables students to fully understand and analyze the relationship between the components of a process and what it can achieve.

- **Recognizing and Defining Computational Problems**
  After completing the CoderZ curriculum, students become more adept at analyzing and breaking down challenges into its parts and solving each area of concern.

- **Testing and Refining**
  The process of creating, testing, and improving upon a solution - both collaboratively and independently - is integral component of the CoderZ methodology. Students systematically identify the source of any problems, and then use an iterative design process which can be easily modified and shared.

- **Fostering an Inclusive Computing Culture**
  CoderZ shows how computational artifacts impact health and wellbeing in both positive and negative ways.

CSTA Standards Mapping

CoderZ maintains standards mapping of each lesson in each curriculum. As the lesson are activity driven, the standards mapping commences in the lesson plan, where the student learning outcomes are defined – as well as which standards they are aligned to:

Objectives

Guiding Question: How can planning help to make better programs?

Learning Objectives and Assessment:

Describe the benefits of planning an algorithm before beginning to code

- In the Reflection Questions, ensure that students understand that planning can help them to understand what the code is supposed to do before coding, saving time and frustration that can come from a "guess and check" method.

Decompose a problem into its components

- Check student success on Mission 7, which requires that they break down the robot path into its component parts.

Students will:

Modify drive blocks to accept a distance parameter
Use measurements to assist with planning a solution

Standards

- CSTA 1B-AP-11 Decompose (break down) problems into smaller, manageable subproblems to facilitate the program development process.

In addition, there is a detailed mapping per activity assigned in the lesson, that specifies the standard, as well the concept being taught.

Cyber Robotics 101 Course	Cyber Robotics 101 Course	Cyber Robotics - CoderZ	Elements of Computational Thinking	Computer Science Teachers Association Standards by Mission
Mission Pack	**Mission**	**Coding Skills**	**CT Elements**	**CSTA - Standard**
Basic Navigation II	mission-Roundabout Left	Motion Planning - direction	Decomposition	1B-AP-10
Basic Navigation II	mission-Roundabout Left	Motion Planning - power	Algorithm Design	2-AP-13
Basic Navigation II	mission-Roundabout Double	Motion Planning - direction	Decomposition	1B-AP-10
Basic Navigation II	mission-Roundabout Double	Motion Planning - power	Algorithm Design	2-AP-13
Basic Navigation II	mission-Roundabout Double	Control Flow - wait for		
Basic Navigation II	mission-Left Curve Turn	Motion Planning - direction	Decomposition	1B-AP-10
Basic Navigation II	mission-Left Curve Turn	Motion Planning - power	Algorithm Design	2-AP-13
Basic Navigation II	mission-Left Curve Turn	Motion Planning - speed		
Basic Navigation II	mission-Right Curve Turn	Motion Planning - direction	Decomposition	1B-AP-10
Basic Navigation II	mission-Right Curve Turn	Motion Planning - speed	Algorithm Design	2-AP-13
Basic Navigation II	mission-Curve Back Left	Motion Planning - direction	Decomposition	1B-AP-10
Basic Navigation II	mission-Curve Back Left	Motion Planning - power	Algorithm Design	2-AP-13

Respondents Targeted Grade Level

CoderZ Module	Targeted Grade Levels
	**2 – 5**

6 – 8

7 – 12

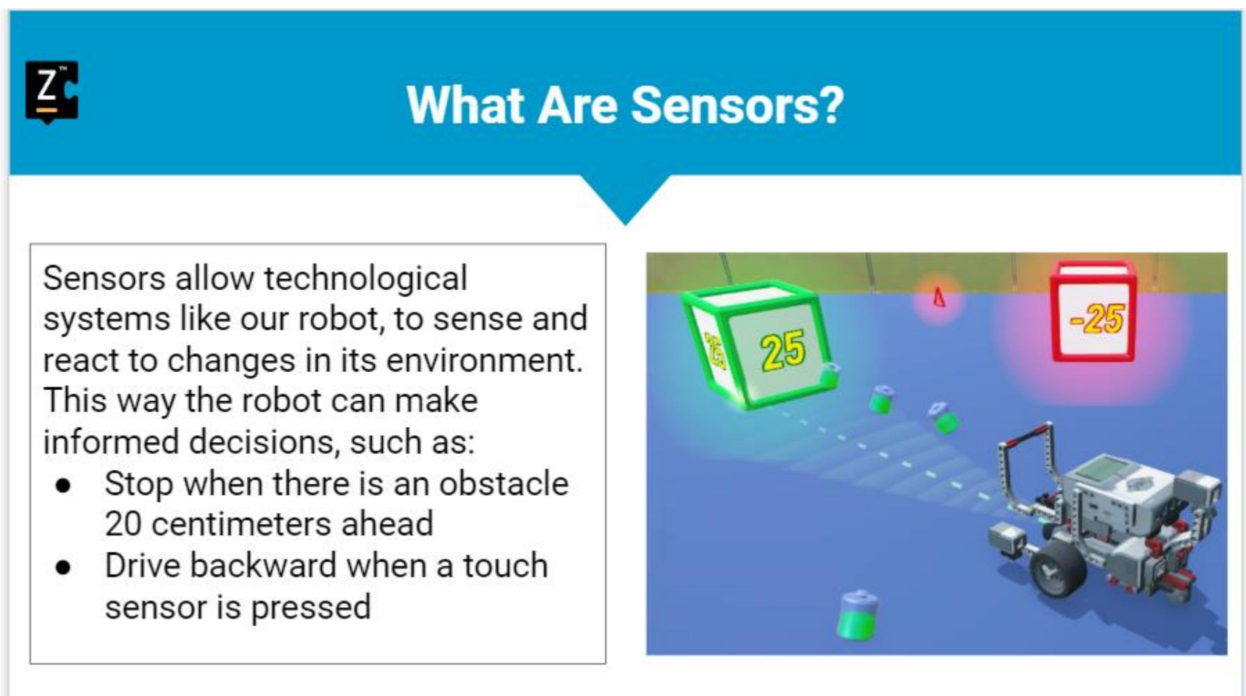
8 – 12

CoderZ LEAGUE -JUNIOR-

5 – 10

8 –12

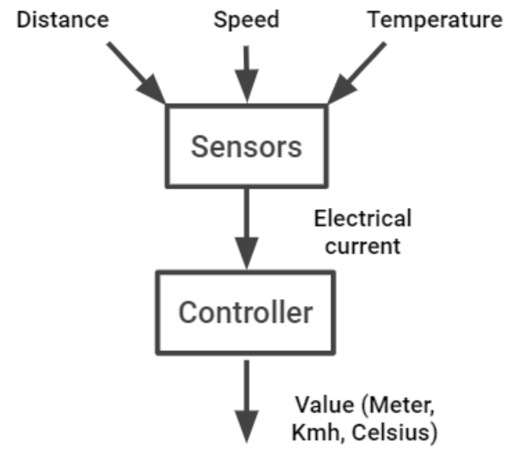
Sample Artifacts from the curriculum



How Do Sensors Work?

Each sensor detects different variables, but the principal is similar in most sensors:

- The sensor measures a physical quantity such as distance or temperature
- The measurement is translated into electrical voltage
- The system controller converts the voltage to a value

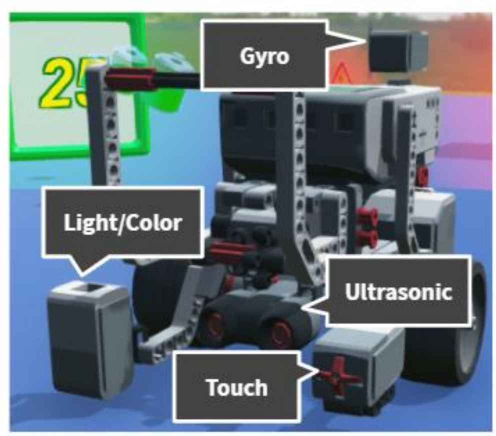


The Sensors

Our robot is equipped with the following sensors:

- Touch - for collision
- Gyroscope - for rotation
- Ultrasonic - for distance
- Light/Color - for surface color/brightness

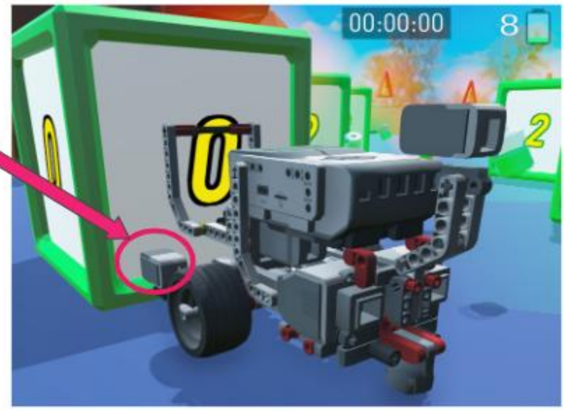
Each motor also has an optical encoder - for motor rotation



The Touch Sensor

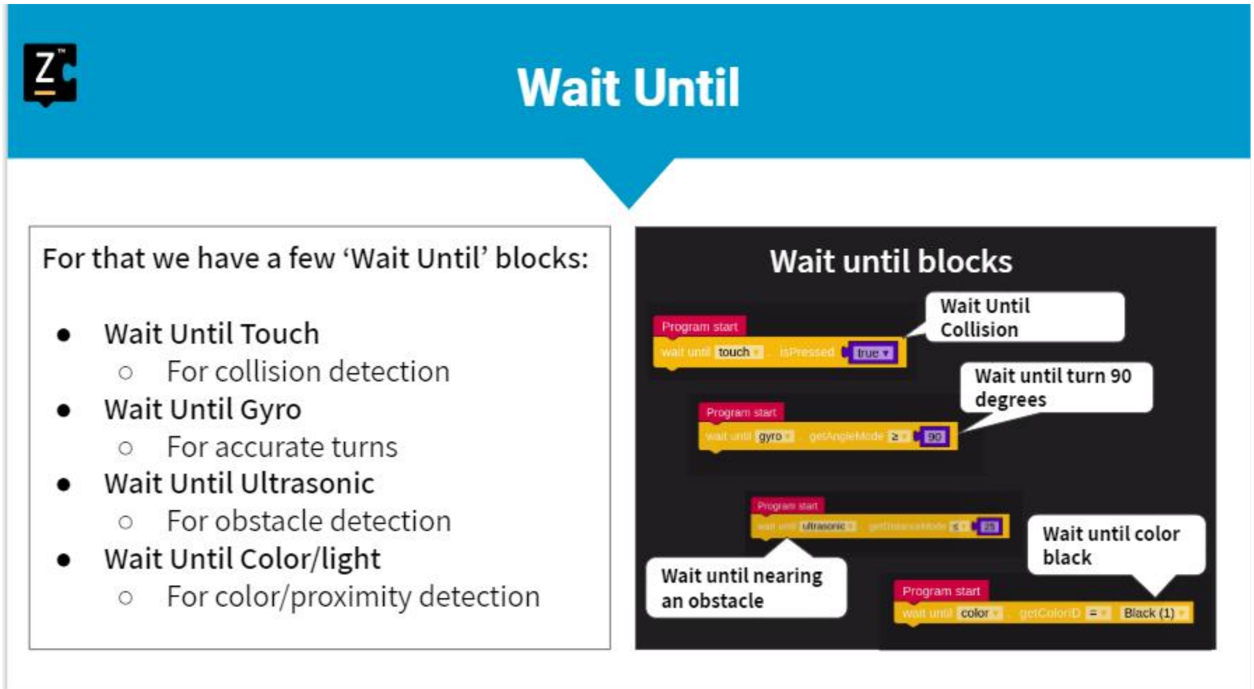
The touch sensor is located at the front of the robot, on its left arm.

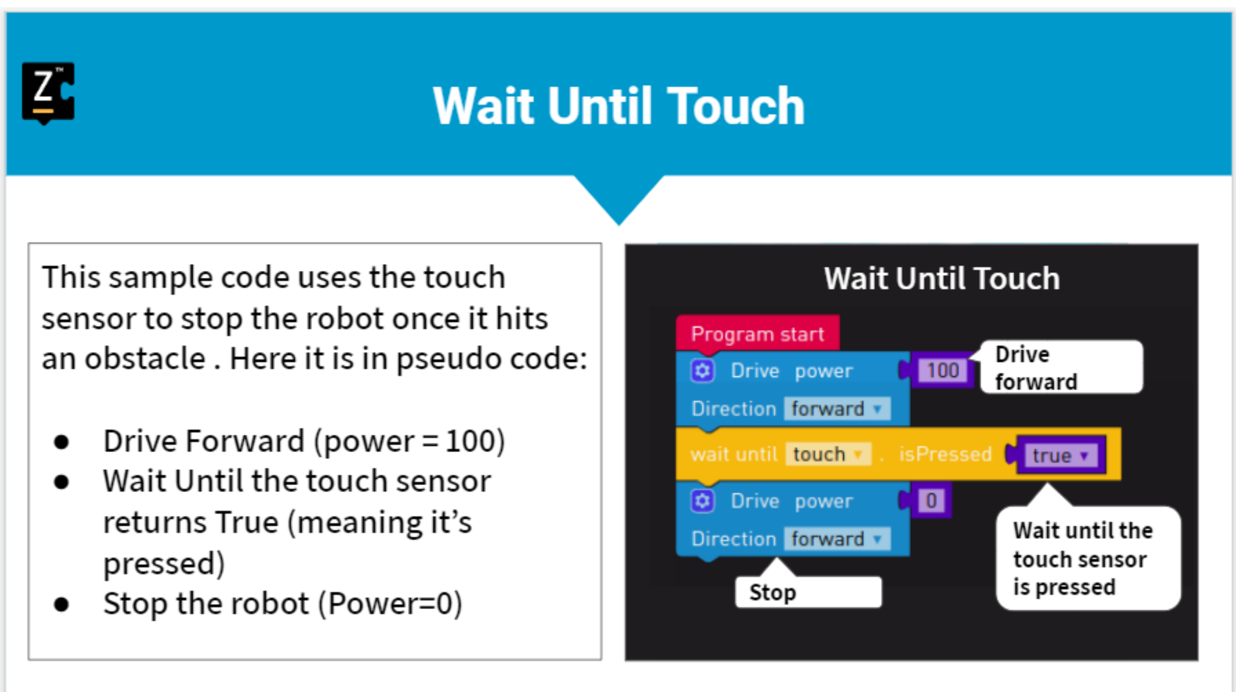
- It is great for collision detection
  - When the sensor is pressed, it returns a boolean value "True"
  - When released, it returns a boolean value "False"



Wait Until

- Sometimes we want the robot to continue until something has happened…
  - Go forward until you see an obstacle
  - Point turn right until you turn 180 degrees
- The sensors feed us with data we can use to control the robot's behavior



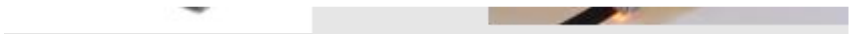


Programming and Robotics Software

- CoderZ is a robotics programming and virtual simulation cloud-based software that uses Blockly, Java and Python programming.
- CoderZ offers participants an intuitive code builder interface that is adaptable for all levels of programming skills
- CoderZ is great for both beginners and experts. Newbies start with our Blockly, Scratch-like programming while more advanced users can go to the object-oriented code editor

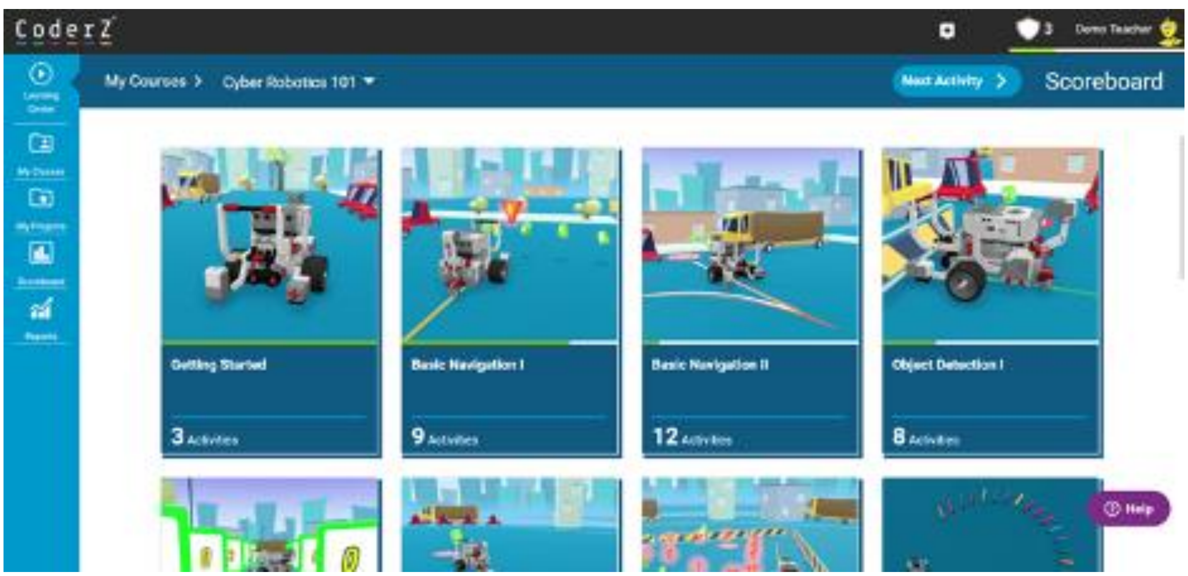
Software Simulation

- Immersive 3D robot simulation - CoderZ intelligent real-time simulation models the behavior of a real robot.
- Students enjoy immediate feedback from the simulation that allows them to code and play with virtual robots outside the robotics lab.
- Can be accessible from anywhere using a standard web browser – to be used from school or from home



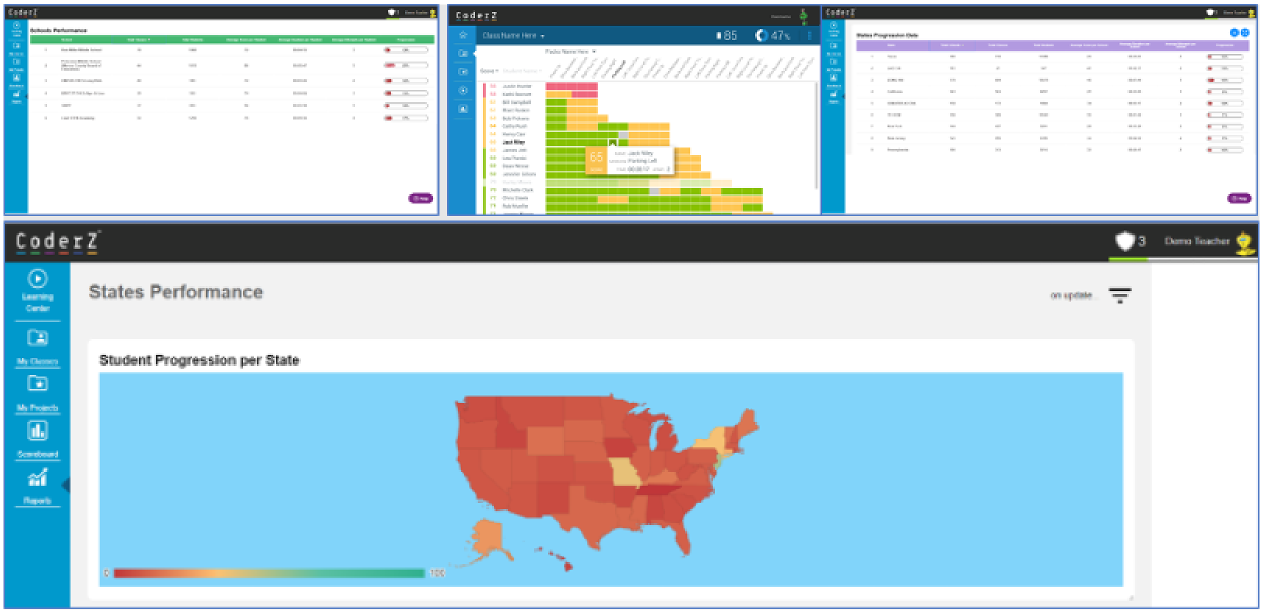
Gamified Missions

- CoderZ includes many gamified Missions in which students learn the basics of coding and robotics by completing challenges through the use of code.
- Missions are great for self-paced learning and gradually build knowledge through hands-on experimentation.
- The missions start at a very basic level and gradually increase in difficulty. Many of the missions can be solved in more than one way, developing problem solving skills and creativity.



Reporting and Measurable Outcomes

- All your data in one place, in real time
- Compare student's current performance with historical baselines or previous programs
- Spot trends, like popular times of day to study, or the students who are ahead or behind the curve
- One intuitive dashboard to follow student progress and achievement



Describe how the curriculum and professional development can fit into a K-12 CS plan, as required in HF 2629.

House File 2629, signed into law in 2020, built on Senate File 274 by requiring computer science instruction. High schools must offer at least one high-quality one-semester course starting July 1, 2022. Middle schools must provide high-quality computer science in seventh or eighth grade by July 1, 2023. Elementary schools must provide high-quality computer science in at least one grade level by July 1, 2023. Schools and the state also must develop K-12 computer science plans by July 1, 2022.

CoderZ developed all courses as age-appropriate points of entry into computer science that sets students up for success, no matter their background or experience. All courses are leveled for experience and may be implemented at elementary, middle, and high school with an average instructional time of 20-25 hours per course perfectly supporting House File 2629 at all levels.

With nearly 100 hours of curriculum and additional extra-curricular activities, such as CoderZ League – the nation's leading cyber robotics competition - CoderZ is both comprehensive and flexible enough to fit into a K-12 plan. They key element to working with CoderZ is that our platform and our people are highly adaptable and willing to work with schools to suite their specific needs. Adoption varies from after school enrichment activities to full scale CSTEM program; some of the most frequent scenarios including:

District Adoption

When planning large scale implementation of a robotics program, CoderZ is a perfect fit. Districts can use the various curriculums offered by CoderZ to different grade levels in the district. By using the CoderZ reporting tools, the district administration can get a clear picture of progression in the district.

- **Exploring STEM**

CoderZ is adopted by schools as one of several programs that are offered to students that will explore several STEM areas during the academic school year. These topics may include: Solar Energy; Electric Vehicles; Video editing and Robotics. In such cases, CoderZ will be used as the curriculum used for exploring robotics.

- **Scaling Robotics Programs**

  Schools with limited budgets for purchasing physical robots will adopt CoderZ and by that enable more students to take robotics. Students can now work in groups, when some of the students carry out projects on the physical robots, while others will focus on developing their computational skills, coding a virtual robot.

- **STEM Electives**

  Certain schools offer different STEM topics as full semester or full year electives. CoderZ supports a progression from introduction to basic concepts in robotics and coding, through to advanced programing of complex algorithms.

- **After School**

  CoderZ is adopted by after school clubs that offer enrichment activities to students in coding and robotics. The activity could be completely virtual or tied to a physical robotics activity. Many of the after-school clubs will also participate in the CoderZ League (competition).

- **Implementation Overview**

  Standing behind our world-class platform are the people dedicated to supporting educators every day. We work with schools every step of the way, from onboarding to training to support, to ensure that you move through the implementation phase as quickly as possible and drive towards full adoption of the platform. It is our goal to see CoderZ become part of the fabric of every school's learning community, so we organize the CoderZ rollout into key phases (customized for every engagement):

  - Onboarding

    Identifying key stakeholders in the rollout process and organizing our game plan, including timelines, milestones, and key deliverables

    - Identifying key technical, training, and communication needs required to move forward

- Implementation
    - Platform activation and account set up
    - Whitelisting web links with IT administration
    - Utilization of Single-Sign-On and rostering platforms, as needed, such as ClassLink, clever, and Google
    - Review of key learning standards and learning objectives to ensure teachers are using a common language and that our shared stakeholders are working in alignment

- Ongoing Training and Support
    - Onboarding webinars customized to your district
    - Designated education success manager focused on delivering training and support to all stakeholders in the district
    - Knowledge hub for self-service support and self-paced training

- Adoption
    - Level-setting expectations for each quarter, semester, and the full-year to ensure we are working towards specific utilization goals
    - Utilization reporting and check-ins, including support and interventions as needed for classrooms that have yet to meet goals

With every school and district engagement, we strive for full adoption. While some platform providers are satisfied with introducing and activating technologies in schools, we are committed to seeing every district can benefit from true adoption of CoderZ.

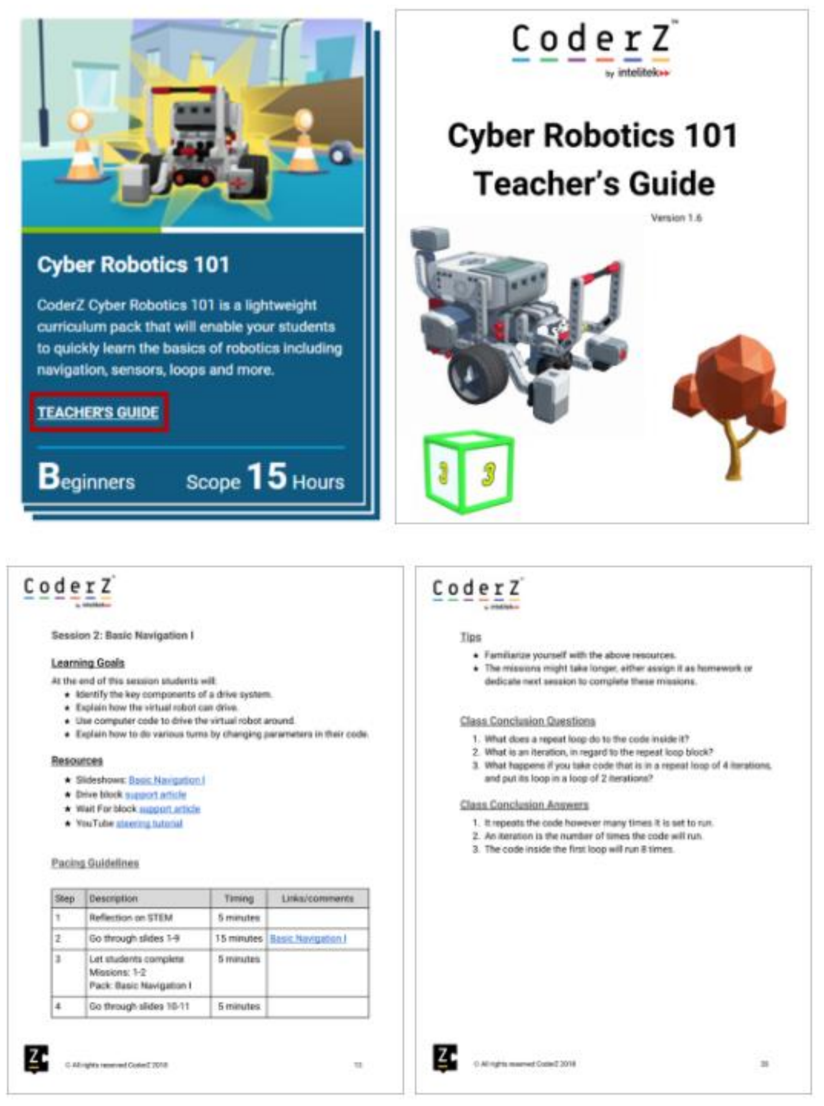
There are no third-party respondents involved in our implementation strategy. Our in-house team works directly with our school customers to ensure strong onboarding, implementation, and adoption. This holds true, as well, for our ongoing support of the platform.

There is no STEM, coding, or robotics expertise or skill set required by personnel from the State of Iowa. CoderZ is designed to be accessible and available to all skill levels, so personnel need only bring their perspective on best-practices for a successful for a rollout of the platform at a state, district, or school level.

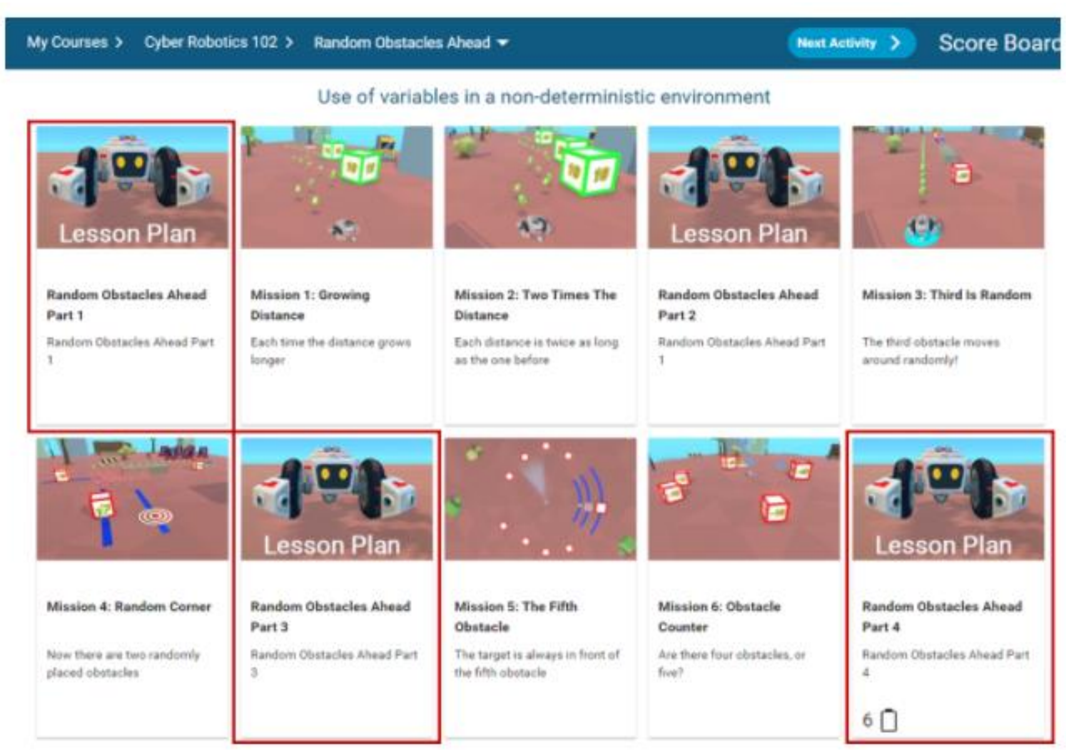
In terms of time requirements, we anticipate initiating a series of virtual meetings with representatives from the state, lasting approximately 60 minutes each, so that our team understands the nuanced requirements of schools and districts in Iowa and to ensure that our work aligns with initiatives that the State is implementing in parallel. We recognize that the State will have many important initiatives running at any given time, as will schools and districts, so we want to make sure that our efforts to foster CoderZ adoption work in concert with the priorities and plans of the State.

In terms of documentation that is provided with the CoderZ platform, we include a robust portfolio of resources and documentation that support the onboarding, implementation, and full adoption of the platform, all at *no additional cost.*

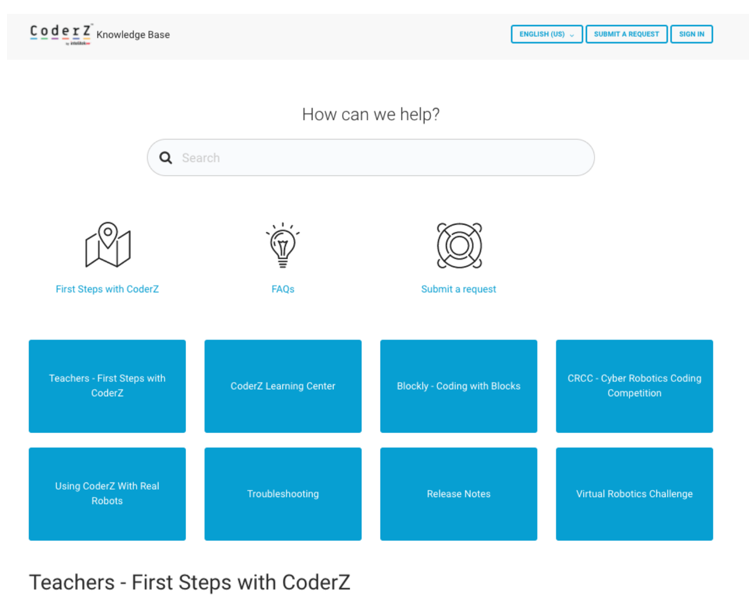
- Every course includes a Teacher's Guide integrated directly into the platform, providing fast, easy access to a critical resource



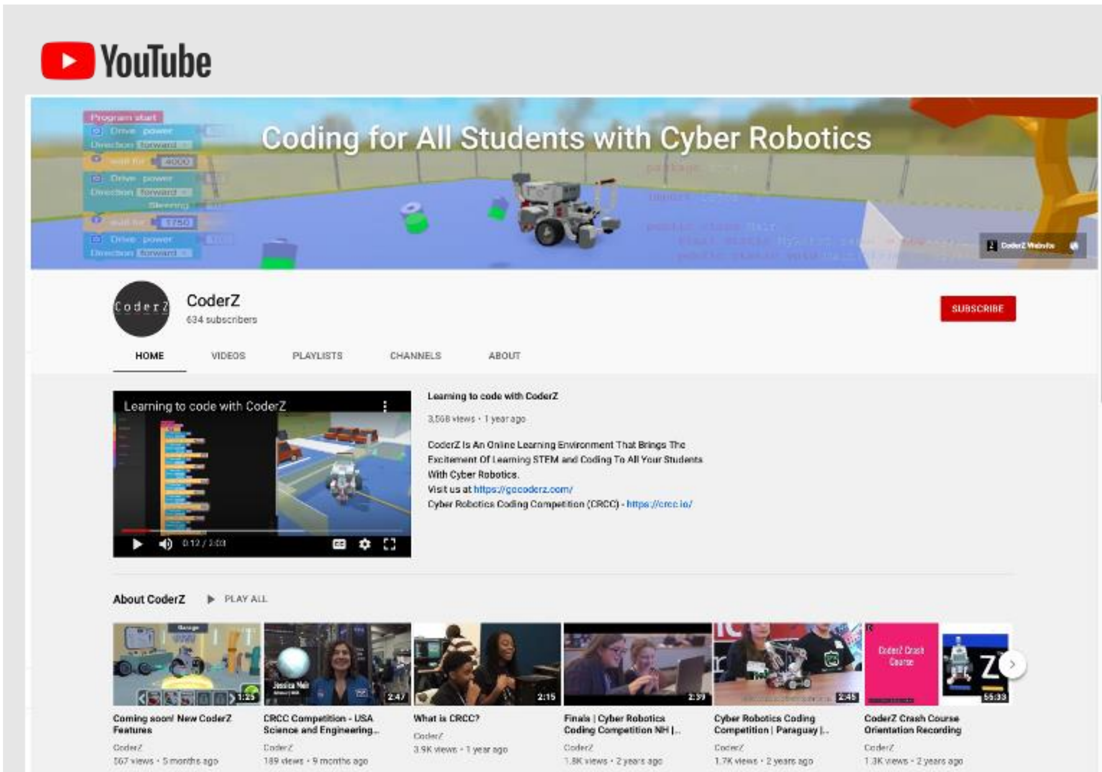
- Teachers have access to ready-made Lesson Plans that help guide instruction and ensure alignment to standards



- Our Knowledge Base is available to all users, offering an in-depth "how-to" that helps teachers and master the platform quickly, ensuring quick implementation and sustained support for all



- We host a popular YouTube channel with ongoing updates and presentations by the CoderZ team as well as by educators who share insights and advice on how to make the most of the platform.



Describe how the curriculum addresses diverse learners, including the gender participation gap, traditionally underrepresented minority students, students with disabilities and English learners.

CoderZ is a web-delivered, standards-aligned, gamified, digital learning environment that supports computer science and STEM with a focus on coding and cyber robotics. The curriculum provides a fun and engaging opportunity to integrate math, physics, engineering, and coding into a variety of learning experiences for ALL students.

To address equity issues, CoderZ incorporates a number of key elements, including:

- Low Floor / High Ceiling: CoderZ is designed to allow easy entry into the virtual learning environment for teachers and students learners with little or no experience in robotics. This “Low Floor” access means that even traditionally marginalized and underrepresented students, many of whom would be reluctant to try robotics due to intimidation from prior experiences or pre-conceived notions about who “should” be interested in this field.

Our most disenfranchised students and learning communities need opportunities to be elevated into STEM experiences. Students who may have otherwise been discouraged due to their inexperience with coding, will be engaged due to the tiered approach of problem-based tasks and missions found within the curriculum.

- Gender Neutrality: The entire universe within which CoderZ robots operate is focused on gender-neutral robots. From naming conventions to color choices, we do not distinguish between girl robots and boy robots specifically to ensure that all learners feel welcome.

- Abilities: Because the CoderZ curriculum enables a 1:1 virtual robotics and coding experience, more students are included, regardless of the opportunity (or lack thereof) to engage with and utilize a physical robot. Additionally, students and teachers with disabilities who may have previously faced barriers in an in-person setting are now able to share their knowledge and strengthen their coding skills while working as an individual or small group from the safety of home – or wherever they are able to log in.

- English Learners: CoderZ understands the importance of including English Language Learners into the field of robotics instruction, which is why the platform is available in English and ten additional languages. CoderZ has been deployed in dozens of countries across five continents, underscoring its ability to reach all learners. That said, another key aspect of supporting the inclusion and development is the embedded professional development, designed to help teachers interact and instruct with confidence in the moment so that they can focus on the individual needs of each learner, including English Language Learners.

By providing a flexible and scalable coding curriculum, able to be utilized and implemented in any setting (in-person, virtual, or hybrid), students gain additional access to opportunities that will impact long-term improved learning outcomes in the coding and robotics space. Access will also help to directly change the life trajectories of the most underserved and underrepresented.

The terms “underserved” and “underrepresented” in this sense is referring to a wide range of people that are underrepresented in STEM professions. Individuals with disabilities, immigrant populations, females, the homeless or displaced, LGBTQ+ populations, and certain racial and ethnic groups often lack role models or clear pathways into the STEM field. It is important that we work, as educators, to lift up all voices to impact the decision-making processes surrounding STEM.

CoderZ is guided by its vision to introduce coding and robotics to as many students and teachers as possible. By doing so, students will have the opportunity to gain knowledge that will shape their ability to make solution-based decisions. Investing in STEM education will provide students positive interaction and collaboration opportunities with their peers. Students will have a greater opportunity to become impactful science and technology leaders who will make strong and innovative decisions for a brighter tomorrow.

Exhibit 14 – Professional Development

Describe the professional development delivered by the Respondent around implementation of the curriculum

CoderZ is an online educational environment that fosters 21st century skills while having fun as they program their own virtual robot. What makes CoderZ so special is that the product and supporting services – including our professional development offerings – help teachers learn and grow in much the same way we help students do the same.

With that said, an education platform is only as powerful as the people behind it, which is why we have developed a professional development roadmap that ensures teachers are equipped to deliver high quality STEM, coding, and robotics instruction. We emphasize core teaching and learning principles that, while anchored to the CoderZ experience, are transferable to all aspects of STEM instruction.

Teacher Resources

Every CoderZ teacher has access to a portfolio of resources specifically designed to set them up for success. Even with no computer science background, teachers can engage in CoderZ materials and professional development quickly and confidently. Our Teacher Resources portfolio continues to expand, and already includes:

- Student outcome plans
- Lesson plans tied to CoderZ modules
- Speaking notes
- Pacing guides
- Questioning techniques, including specific questions for use in the classroom
- Discussion and writing prompts
- Assessment guides, including how to use progress monitoring for formative assessment

Teacher Professional Development Course Overview

Teachers in districts that adopt CoderZ will have immediate access to our standard Professional Development course, which includes an initial 2 - 3 hours of free, embedded self-guided training (additional scheduled webinars, etc... TBD depending on the scope of implementation), that sets them on the path to teaching computer science curriculum well within the 6 – 12-month time frame noted in this RFP.

Additional "paid PD" services can be customized depending on courses and teacher experience in coding/robotics content (see Cost Proposal for additional detail regarding pricing).

Our goal is that after participation in an initial two (2) hour of Professional Development session, teachers are ready to begin implementing their assigned CoderZ courses with students immediately. Depending on the CoderZ course, frequency of scheduling, and time devoted to implementing the program, follow up Professional Development sessions may be scheduled to begin within 4-8 weeks after starting the program. We aim to provide relevant just in time training resources, both self-guided and instructor led, to guide teacher through the delivery of our curricula with their students. Total time devoted to Professional Development will vary based on the level of course complexity and attendee's prior experience level with coding and robotics concepts. A target of between 6 and 8 hours of Professional Development is recommended (but not required) to accompany any single CoderZ course through the first year of implementation.

Our course work includes, but is not limited to:

- An introduction to Computer Science principles and the world of educational robotics
- Pedagogy of discovery-based learning in Computer Science
- Teaching strategies for blended learning
- Integrating math and science into problem solving in Computer Science
- Leveraging student engagement data and reporting tools to apply personalized teaching interventions
- Strategies for teaching in a gamified environment that maintain validity and reliability
- Incorporating key 21st-Century skills into Computer Science instruction, including:
    - Collaboration
    - Computational thinking
    - Effective listening
    - Engaging in discussions
    - Giving and receiving constructive feedback further develop ideas
- An introduction and overview of the CoderZ online curriculum, including how the curriculum maps to CSTA and State standards, how it can be applied in traditional classrooms, hybrid learning scenarios, or virtual learning environments.
- Simulation and practice with various teaching environments and classroom scenarios.

Implementation of our professional development is designed to be flexible and can be delivered on-site, online, or a hybrid of both, though we anticipate utilizing fully online delivery of our professional development courses in Iowa at least through the 2020-2021 school year. We are mindful of the precautions required by COVID-19 and view this as a cost-saving maneuver for districts without sacrificing quality.

By the time teachers have completed our coursework and brought students through the first missions of CoderZ, they leave with a deep understanding of computer science fundamentals, even if they bring no experience into the work at the outset. Further, teachers can immediately apply fundamental concepts in the classroom (in-person or online) with the support of our instructional guides, teacher resources, formative assessments, and reporting tools.

Implementation Overview

Standing behind our world-class platform are the people dedicated to supporting educators every day. We work with schools every step of the way, from onboarding to training to support, to ensure that you move through the implementation phase as quickly as possible and drive towards full adoption of the platform. It is our goal to see CoderZ become part of the fabric of every school's learning community, so we organize the CoderZ rollout into key phases (customized for every engagement):

- Onboarding
    - Identifying key stakeholders in the rollout process and organizing our game plan, including timelines, milestones, and key deliverables
    - Identifying key technical, training, and communication needs required to move forward
- Implementation
    - Platform activation and account set up
    - Whitelisting web links with IT administration
    - Utilization of Single-Sign-On and rostering platforms (currently Clever, ClassLink, and Google)
    - Review of key learning standards and learning objectives to ensure teachers are using a common language and that our shared stakeholders are working in alignment
- Ongoing Training and Support
    - Onboarding webinars customized to your district
    - Designated education success manager focused on delivering training and support to all stakeholders in the district
    - Knowledge hub for self-service support and self-paced training
- Adoption
    - Level-setting expectations for each quarter, semester, and the full-year to ensure we are working towards specific utilization goals
    - Utilization reporting and check-ins, including support and interventions as needed for classrooms that have yet to meet goals

With every school and district engagement, we strive for full adoption. While some platform providers are satisfied with introducing and activating technologies in schools, we are committed to seeing every district can benefit from true adoption of CoderZ.

There are no third-party respondents involved in our implementation strategy. Our in-house team works directly with our school customers to ensure strong onboarding, implementation, and adoption. This holds true, as well, for our ongoing support of the platform.

Describe the ability to prepare teachers to teach the curriculum within 6-12 months

As noted, and outlined above, our goal is that after participation in an initial Professional Development session, teachers are ready to begin implementing their assigned CoderZ courses with students immediately. Depending on the CoderZ course, frequency of scheduling, and time devoted to implementing the program, follow up Professional Development sessions may be scheduled to begin within 4-8 weeks after starting the program. We aim to provide relevant just in time training resources, both self-guided and instructor led, to guide teacher through the delivery of our curricula with their students. Total time devoted to Professional Development will vary based on the level of course complexity and attendee's prior experience level with coding and robotics concepts. A target of between 6 and 8 hours of Professional Development should be dedicated for any single CoderZ course through the first year of implementation.

Provide sample artifacts from the professional learning:

SAMPLE Course Outline that could be utilized during a professional learning opportunity:

Course Outline

Session Format

Learning Goals	Each session focuses on new learning goals, while building on top of previous experiences and lessons learned. We outline at the top of each session the learning goals.
Resources	We have developed different resources for each session, for you to use in class. These resources include presentations (see below), how-to articles and video tutorials. Familiarize yourself with them prior to each session.
Pacing Guidelines	The pacing guidelines will let you know how the lesson is divided between presentations, discussions and CoderZ missions. Remember that timing varies from class to class so remember these are guidelines, not specific instructions to follow. If you class needs more time, let them. If they progress faster than expected, reflect (see below) on their progress.
The Presentation	Each session is guided by a presentation that covers the entire session from theory to practice, including: > Scientific background > Examples and sample code > Discussion guidelines > Play time activities > Speaker notes The presentation shows when and which missions to complete so all you'll need in front of you are the slides and our speaker notes.

SAMPLE Lesson that could be utilized during a Professional Learning opportunity:

NOTE: Live Links can be inserted and utilized during sessions to provide access to additional resources during the learning process

Lesson 1: Intro to STEM & CoderZ

Learning Goals

By the end of this lesson students will:

- Understand the meaning of STEM, and how the subjects are integrated.
- Have a basic comprehension of what robots are.
- Analyze technological systems based on their inputs and outputs.
- Login to CoderZ and complete basic navigation missions.

Resources

- Slideshow: Intro to STEM & CoderZ
- Drive block support article

Pacing Guidelines

Step	Description	Timing	Links/comments
1	Go through the slideshow slides 1-12	20 minutes	Intro to STEM & CoderZ
2	Let students sign up to CoderZ	5 minutes	Using your CoderZ class link
3	Let students complete the Getting Started pack	5 minutes	3 missions
4	Complete intro conclusion activity + Class Conclusion Questions	15 minutes	Slide 13

SAMPLE questioning strategies to be utilized during Professional Learning opportunities:

CoderZ™

The Touch Sensor

Think of 4 different touch sensors that you use in your everyday life.

Consider where they are located.

Journal about each of the touch sensors that you thought of:

What is its purpose?

If you would be the designer of it, how would you design it to best fit your needs?

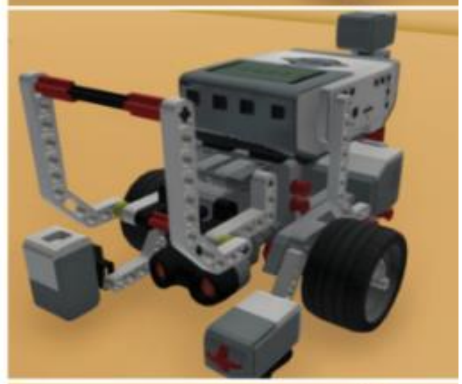
Why in your opinion the touch sensor in metter, was designed as it is now?

Discuss what you just journaled with your pair.

In groups of 4, discuss the following:

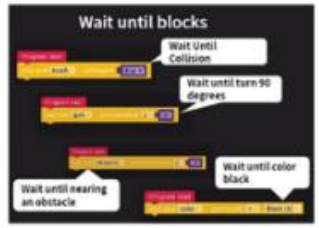
- What material goals did this discussion cover?
- What pedagogical goals, did this activity cover?
- How does these questions, serve the learning of the touch sensor?

Come up with 3 different questions, that can have the same learning outcomes, and pedagogical outcomes we talk about.



CoderZ™

Wait Until



Think back on three different occasions when you needed something to continue and happen until it met a condition (like whipping cream until it is stable).

In your group:

Come up with four different designs and ideas of a block in Blockly an instruction block, that will mean:

do not move on if the condition is not met.

Make use of as few words as possible, and of parameters.

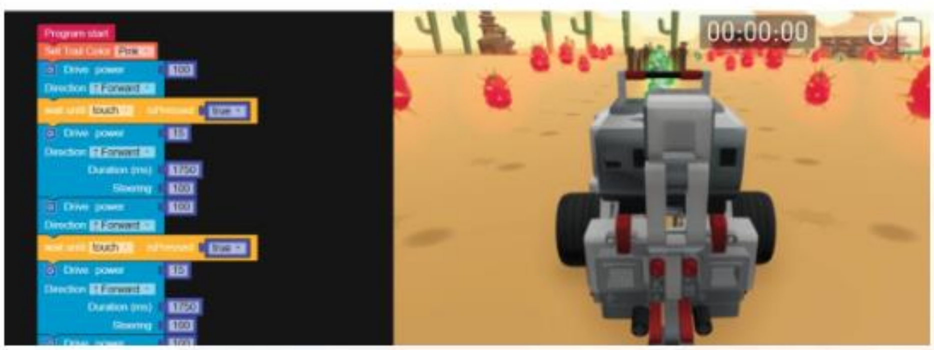
Check in CoderZ for the Until Blocks.

Whole group discussion:

- Why were these Blocks used?
- What might be the difficulties for the students in understanding these blocks meaning?

In groups of four, come up with few ideas to facilitate these instructions.

Keep in mind to use more than one pedagogical strategy to get to as many students as possible.



SAMPLE student learning outcomes can be shared during Professional Learning opportunities which could help provide significant guidance to the teacher regarding specific lessons:

Objectives

Guiding Question: How can planning help to make better programs?

Learning Objectives and Assessment:

Describe the benefits of planning an algorithm before beginning to code

- In the Reflection Questions, ensure that students understand that planning can help them to understand what the code is supposed to do before coding, saving time and frustration that can come from a "guess and check" method.

Decompose a problem into its components

- Check student success on Mission 7, which requires that they break down the robot path into its component parts.

Students will:

Modify drive blocks to accept a distance parameter
Use measurements to assist with planning a solution

Standards

- CSTA 1B-AP-11 Decompose (break down) problems into smaller, manageable subproblems to facilitate the program development process.

ADDITIONAL OVERVIEW OF SAMPLE ARTIFACTS that could be utilized during Professional Learning Opportunities:

The **teacher guide** provides explanations and instructional resources relevant to each lesson and the activities assigned therein. The text sample below demonstrates how our professional development methodology strives for relevance in the classroom not only in its subject matter, but in its tone and delivery. It's critical that we get into the hearing of teachers so that they can enter into the work with great confidence and enthusiasm.

Sample: Mission 6

The mission is significantly more difficult from the ones the students have seen so far for two reasons: it involves more steps and students are asked to write all their code from scratch, rather than just arranging given blocks. The mission also introduces a new game element, moving platforms controlled by buttons. You should have your class discuss and work on the first part of this mission together, before allowing students to complete it on their own.

Let students watch the tutorial for this mission. It explains that the button needs to be pressed to create a bridge to reach the target but does not give explicit instruction for solving the mission. Remind students of the discussion from the introduction about breaking down a problem into subproblems.

Let students **think** and journal for 60sec: How can the mission be broken down into two smaller problems?

Now let the students **pair** with one of their peers (split into pairs either using the virtual session's capabilities, or in the class) (Each student gets 60 seconds to explain her Idea, and the other - listens. They switch roles after 60 sec)

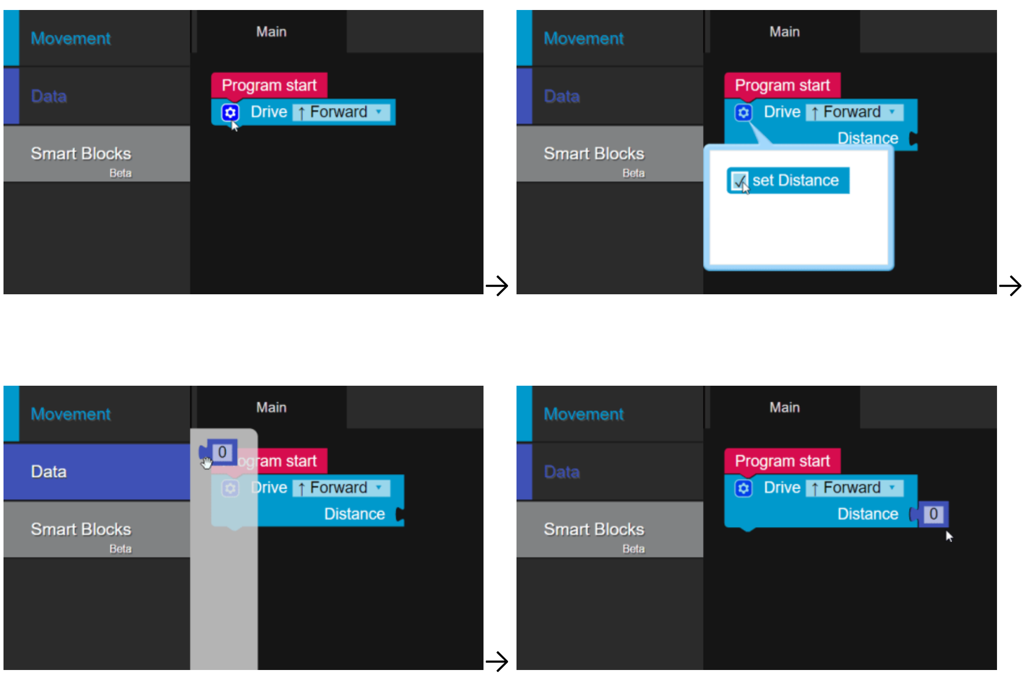
Share the thoughts. Let each student share their idea.

Something like this might come up:

- Get the robot from the starting point to the button
- Get the robot from the button to the target

Now ask students to break that first step down further. What does the robot need to do to get from the starting position to the button? Pretty clearly it needs to drive a distance, turn, and drive a little more. But how far to drive each time. Have students switch to Explore Mode. Not only does this top-down view let them see the entire environment, it also reveals some annotations that have been added to show the important distances. From this they should see that the first step the robot needs to take is to drive forward 8 meters.

Since the default Drive block available in the menu is not ready for a distance parameter, students may need a reminder of how to add one. Click on the  button, check "set Distance", and drag and attach a number block from the Data menu.



At this point students should be in a position to write the three steps necessary to get the robot to the button (Drive 8; Turn Left; Drive 2). Let them do so and then test their solutions by running the simulation. Once they have successfully navigated the robot to the button, they can then work on getting the robot from the button to the target. Remind them that they need to press the Reset Simulation button between tests, so the robot returns to its starting position.

If anyone needs help with the second part of the program you can point out that when the robot reaches the button the target is behind it. That means the robot will either need to turn around or move backwards to teach the target. Students should review missions 4 and 5 if they need a refresher on turning around and moving backwards.

One other tip that will be helpful to everyone: If you duplicate (using the right-click menu) a block that is already configured with a distance parameter, the new block will have the parameter as well. That can be an easier way to add multiple Drive distance blocks than building each one separately.

On-Demand Professional Development Examples:

In addition to a growing library of teacher resources and our professional development courses, we deliver on-demand webinars, provide short "just-in-time" video resources, help and support topics (with embedded learning videos), etc...

We recruit from within our ranks as well as from a global network of expert educators and technologists, each of whom bring timely, relevant training directly to educators.

Here is an Example: Including embedded Teacher PD Resources (Free) - built into lesson plans teacher preparation – including embedded video links

TOPIC: What are Smart Blocks:

What are Smart Blocks?

CODERZ SUPPORT TEAM

Smart Blocks work for EV3, SPIKE™, Zappi, and Ruby!

Smart Blocks are blockly functions that you can create for yourself to be used in any other project or mission. They can include any code you like, preferably snippets of code you use often in many CoderZ missions and/or projects.

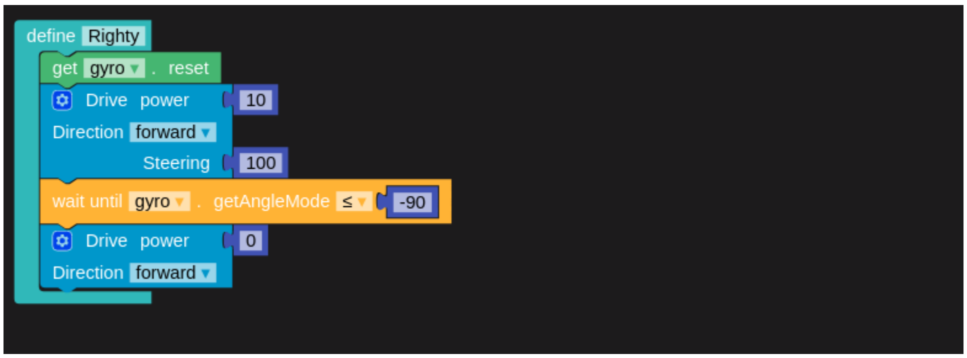
Tips and Tricks:

- Once you create a Smart Block, it becomes a part of your profile and will go with you wherever you go.

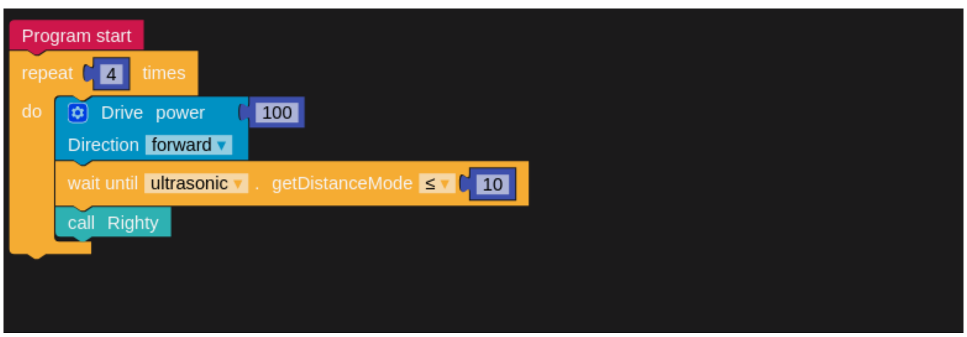
- You can also [import existing Smart Blocks]() to your code for easy access.
- Editing, changing, clearing or deleting Smart Blocks in one mission or project will result in the same action being performed throughout your user. Tread with caution!

Check out the video tutorial on YouTube!

In the example below we created a Smart Block that will make the robot turn 90 degrees to the right.



Here it is being used in code, that drives the robot in a square:



ADDITIONAL EXAMPLE: "Just-in-Time" Instructional Video: **HOW TO CREATE A CLASS**

ADDITIONAL EXAMPLE: How to utilize **ADDITIONAL TEACHER RESOURCES**

ADDITIONAL ON-DEMAND WEBINARS AND RESOURCES – including Professional Learning Opportunities can be found and previewed here: **https://gocoderz.com/blog/online-professional-development/**

Provide an agenda for one day of professional development

Basic Onboarding Agenda:

"Welcome to CoderZ Introductory Training Agenda"

1) **Welcome & Introductions**
2) **Session Overview & Goals**
    a) Ready to implement with students upon completion
3) **Icebreaker Discussion Questions: Coding & Robotics experience**
    a) Understand participant experience level
4) **Importance of Coding & Robotics in future career**
    a) *Meet the Robots at Amazon:* video and discussion
    b) Skills necessary for careers of the future
5) **Introduction to Intelitek, our mission, and CoderZ platform**
    a) History supporting career readiness in robotics/automation
    b) Making robotics and CS education accessible to any student
6) **CoderZ tour, demonstration, and hands on experience**
    a) Understand platform interface
    b) Show increasing levels of content complexity: basic/advanced navigation
    c) Hands on task – Repeat Loops
7) **Teacher Resources overview**
    a) Teachers Guide: slideshows, support articles, video tutorials
    b) Researching topics in the Knowledge Base
8) **Program Management Task**
    a) Account creation
    b) Setting up classes & course assignment options
    c) Basic reporting options
9) **Review Goals and Q&A**

Describe time needed for professional development: length, frequency, availability, and format of training (I.e. online, blended, etc.)

As previously outlined via the implementation overview above (length, frequency, availability, and format of training - i.e. online, blended, etc...), implementation of our professional development is designed to be flexible and can be delivered on-site, online, or a hybrid of both, *though we anticipate utilizing fully online delivery* of our professional development courses in Iowa at least through the 2020-2021 school year. We are mindful of the precautions required by COVID-19 and view this as a cost-saving maneuver for districts without sacrificing quality.

Once again, depending on the CoderZ course, frequency of scheduling, and time devoted to implementing the program, follow-up Professional Development sessions may be scheduled to begin within 4-8 weeks after starting the program. We aim to provide relevant, just-in-time training resources, both self-guided and instructor led, to guide teachers through the delivery of our curricula with their students. Total time devoted to Professional Development will vary based on the level of course complexity and attendee's prior experience level with coding and robotics concepts. A target of between 6 and 8 hours of Professional Development is recommended (but not required) to accompany any single CoderZ course through the first year of implementation.

By the time teachers have completed our coursework and brought students through the first missions of CoderZ, they leave with a deep understanding of computer science fundamentals, even if they bring no experience into the work at the outset. Further, teachers can immediately apply fundamental concepts in the classroom (in-person or online) with the support of our instructional guides, teacher resources, formative assessments, and reporting tools.

Main Criteria: CoderZ Adventure with Lego

Secondary Criteria: Science, Technology, Engineering & Mathematics (CCTC), Iowa Student Standards

Subjects: Career Technical Education, Mathematics, Science, Technology Education

Grades: K, 1, 2, 3, 4, 5

CoderZ Adventure with Lego

Adventure with Lego: Lesson 1: Adventure Peak

Iowa Student Standards

Mathematics

Grade **K** - Adopted: **2012**

STRAND / COURSE	IA.CC.K.MD.	Measurement and Data
ESSENTIAL CONCEPT AND/OR SKILL		Describe and compare measurable attributes.
DETAILED DESCRIPTOR	K.MD.1.	Describe measurable attributes of objects, such as length or weight. Describe several measurable attributes of a single object. (K.MD.1.) (DOK 2)

Iowa Student Standards

Mathematics

Grade **2** - Adopted: **2012**

STRAND / COURSE	IA.CC.2.MD.	Measurement and Data
ESSENTIAL CONCEPT AND/OR SKILL		Measure and estimate lengths in standard units.
DETAILED DESCRIPTOR	2.MD.1.	Measure the length of an object by selecting and using appropriate tools such as rulers, yardsticks, meter sticks, and measuring tapes. (2.MD.1.) (DOK 1)
STRAND / COURSE	IA.CC.2.MD.	Measurement and Data
ESSENTIAL CONCEPT AND/OR SKILL		Represent and interpret data.
DETAILED DESCRIPTOR	2.MD.9.	Generate measurement data by measuring lengths of several objects to the nearest whole unit, or by making repeated measurements of the same object. Show the measurements by making a line plot, where the horizontal scale is marked off in whole-number units. (2.MD.9.) (DOK 2)

Iowa Student Standards

Mathematics

Grade **4** - Adopted: **2012**

STRAND / COURSE	IA.CC.4.MD.	Measurement and Data
ESSENTIAL CONCEPT AND/OR SKILL		Geometric measurement: understand concepts of angle and measure angles.
DETAILED DESCRIPTOR	4.MD.5.	Recognize angles as geometric shapes that are formed wherever two rays share a common endpoint, and understand concepts of angle measurement:
GRADE LEVEL EXPECTATION	4.MD.5.b.	An angle that turns through n one-degree angles is said to have an angle measure of n degrees. (4.MD.5.) (DOK 1)
STRAND / COURSE	IA.CC.4.MD.	Measurement and Data
ESSENTIAL CONCEPT AND/OR SKILL		Geometric measurement: understand concepts of angle and measure angles.
DETAILED DESCRIPTOR	4.MD.6.	Measure angles in whole-number degrees using a protractor. Sketch angles of specified measure. (4.MD.6.) (DOK 1)

Iowa Student Standards

Science

Grade **K** - Adopted: **2015**

STRAND / COURSE	IA.K-PS2.	Motion and Stability: Forces and interactions
ESSENTIAL CONCEPT AND/OR SKILL		Students who demonstrate understanding can:
DETAILED DESCRIPTOR	K-PS2-1.	Plan and conduct an investigation to compare the effects of different strengths or different directions of pushes and pulls on the motion of an object.
STRAND / COURSE	IA.K-2-ETS1.	Engineering Design
ESSENTIAL CONCEPT AND/OR SKILL		Students who demonstrate understanding can:
DETAILED DESCRIPTOR	K-2-ETS1-1.	Ask questions, make observations, and gather information about a situation people want to change to define a simple problem that can be solved through the development of a new or improved object or tool.
DETAILED DESCRIPTOR	K-2-ETS1-2.	Develop a simple sketch, drawing, or physical model to illustrate how the shape of an object helps it function as needed to solve a given problem.

Iowa Student Standards

Science

Grade **1** - Adopted: **2015**

STRAND / COURSE	IA.K-2-ETS1.	Engineering Design
ESSENTIAL CONCEPT AND/OR SKILL		Students who demonstrate understanding can:
DETAILED DESCRIPTOR	K-2-ETS1-1.	Ask questions, make observations, and gather information about a situation people want to change to define a simple problem that can be solved through the development of a new or improved object or tool.
DETAILED DESCRIPTOR	K-2-ETS1-2.	Develop a simple sketch, drawing, or physical model to illustrate how the shape of an object helps it function as needed to solve a given problem.

Iowa Student Standards

Science

Grade **2** - Adopted: **2015**

STRAND / COURSE	IA.2-PS1.	Matter and its Interactions
ESSENTIAL CONCEPT AND/OR SKILL		Students who demonstrate understanding can:
DETAILED DESCRIPTOR	2-PS1-2.	Analyze data obtained from testing different materials to determine which materials have the properties that are best suited for an intended purpose.
DETAILED DESCRIPTOR	2-PS1-3.	Make observations to construct an evidence-based account of how an object made of a small set of pieces can be disassembled and made into a new object.
STRAND / COURSE	IA.K-2-ETS1.	Engineering Design
ESSENTIAL CONCEPT AND/OR SKILL		Students who demonstrate understanding can:
DETAILED DESCRIPTOR	K-2-ETS1-1.	Ask questions, make observations, and gather information about a situation people want to change to define a simple problem that can be solved through the development of a new or improved object or tool.
DETAILED DESCRIPTOR	K-2-ETS1-2.	Develop a simple sketch, drawing, or physical model to illustrate how the shape of an object helps it function as needed to solve a given problem.

Iowa Student Standards

Science

Grade **3** - Adopted: **2015**

STRAND / COURSE	IA.3-PS2.	Motion and Stability: Forces and Interactions

ESSENTIAL CONCEPT AND/OR SKILL		Students who demonstrate understanding can:
DETAILED DESCRIPTOR	3-PS2-2.	Make observations and/or measurements of an object's motion to provide evidence that a pattern can be used to predict future motion.
STRAND / COURSE	IA.3-5-ETS1.	Engineering Design
ESSENTIAL CONCEPT AND/OR SKILL		Students who demonstrate understanding can:
DETAILED DESCRIPTOR	3-5-ETS1-1.	Define a simple design problem reflecting a need or a want that includes specified criteria for success and constraints on materials, time, or cost.
DETAILED DESCRIPTOR	3-5-ETS1-2.	Generate and compare multiple possible solutions to a problem based on how well each is likely to meet the criteria and constraints of the problem.
DETAILED DESCRIPTOR	3-5-ETS1-3.	Plan and carry out fair tests in which variables are controlled and failure points are considered to identify aspects of a model or prototype that can be improved.

Iowa Student Standards

Science

Grade **4** - Adopted: **2015**

STRAND / COURSE	IA.3-5-ETS1.	Engineering Design
ESSENTIAL CONCEPT AND/OR SKILL		Students who demonstrate understanding can:
DETAILED DESCRIPTOR	3-5-ETS1-1.	Define a simple design problem reflecting a need or a want that includes specified criteria for success and constraints on materials, time, or cost.
DETAILED DESCRIPTOR	3-5-ETS1-2.	Generate and compare multiple possible solutions to a problem based on how well each is likely to meet the criteria and constraints of the problem.
DETAILED DESCRIPTOR	3-5-ETS1-3.	Plan and carry out fair tests in which variables are controlled and failure points are considered to identify aspects of a model or prototype that can be improved.

Iowa Student Standards

Science

Grade **5** - Adopted: **2015**

STRAND / COURSE	IA.3-5-ETS1.	Engineering Design
ESSENTIAL CONCEPT AND/OR SKILL		Students who demonstrate understanding can:
DETAILED DESCRIPTOR	3-5-ETS1-1.	Define a simple design problem reflecting a need or a want that includes specified criteria for success and constraints on materials, time, or cost.
DETAILED DESCRIPTOR	3-5-ETS1-2.	Generate and compare multiple possible solutions to a problem based on how well each is likely to meet the criteria and constraints of the problem.
DETAILED DESCRIPTOR	3-5-ETS1-3.	Plan and carry out fair tests in which variables are controlled and failure points are considered to identify aspects of a model or prototype that can be improved.

Iowa Student Standards

Technology Education

Grade **K** - Adopted: **2018**

STRAND / COURSE		CSTA K-12 Computer Science Standards
ESSENTIAL CONCEPT AND/OR SKILL	CSTA.1A.	Level 1A (Ages 5-7)
DETAILED DESCRIPTOR	1A-CS.	Computing Systems
GRADE LEVEL EXPECTATION		Devices
EXAMPLE	1A-CS-01.	Select and operate appropriate software to perform a variety of tasks, and recognize that users have different needs and preferences for the technology they use. (P1.1)
STRAND / COURSE		CSTA K-12 Computer Science Standards

ESSENTIAL CONCEPT AND/OR SKILL	CSTA.1A.	Level 1A (Ages 5-7)
DETAILED DESCRIPTOR	1A-AP.	Algorithms & Programming
GRADE LEVEL EXPECTATION		Program Development
EXAMPLE	1A-AP-12.	Develop plans that describe a program's sequence of events, goals, and expected outcomes. (P5.1, P7.2)

Iowa Student Standards

Technology Education

Grade **1** - Adopted: **2018**

STRAND / COURSE		CSTA K-12 Computer Science Standards
ESSENTIAL CONCEPT AND/OR SKILL	CSTA.1A.	Level 1A (Ages 5-7)
DETAILED DESCRIPTOR	1A-CS.	Computing Systems
GRADE LEVEL EXPECTATION		Devices
EXAMPLE	1A-CS-01.	Select and operate appropriate software to perform a variety of tasks, and recognize that users have different needs and preferences for the technology they use. (P1.1)
STRAND / COURSE		CSTA K-12 Computer Science Standards
ESSENTIAL CONCEPT AND/OR SKILL	CSTA.1A.	Level 1A (Ages 5-7)
DETAILED DESCRIPTOR	1A-AP.	Algorithms & Programming
GRADE LEVEL EXPECTATION		Program Development
EXAMPLE	1A-AP-12.	Develop plans that describe a program's sequence of events, goals, and expected outcomes. (P5.1, P7.2)

Iowa Student Standards

Technology Education

Grade **2** - Adopted: **2018**

STRAND / COURSE		CSTA K-12 Computer Science Standards
ESSENTIAL CONCEPT AND/OR SKILL	CSTA.1A.	Level 1A (Ages 5-7)
DETAILED DESCRIPTOR	1A-CS.	Computing Systems
GRADE LEVEL EXPECTATION		Devices
EXAMPLE	1A-CS-01.	Select and operate appropriate software to perform a variety of tasks, and recognize that users have different needs and preferences for the technology they use. (P1.1)
STRAND / COURSE		CSTA K-12 Computer Science Standards
ESSENTIAL CONCEPT AND/OR SKILL	CSTA.1A.	Level 1A (Ages 5-7)
DETAILED DESCRIPTOR	1A-AP.	Algorithms & Programming
GRADE LEVEL EXPECTATION		Program Development
EXAMPLE	1A-AP-12.	Develop plans that describe a program's sequence of events, goals, and expected outcomes. (P5.1, P7.2)

Iowa Student Standards

Technology Education

Grade **3** - Adopted: **2018**

STRAND / COURSE		CSTA K-12 Computer Science Standards
ESSENTIAL CONCEPT AND/OR SKILL	CSTA.1B.	Level 1B (Ages 8-11)
DETAILED DESCRIPTOR	1B-AP.	Algorithms & Programming
GRADE LEVEL EXPECTATION		Variables
EXAMPLE	1B-AP-09.	Create programs that use variables to store and modify data. (P5.2)

STRAND / COURSE		CSTA K-12 Computer Science Standards
ESSENTIAL CONCEPT AND/OR SKILL	CSTA.1B.	Level 1B (Ages 8-11)
DETAILED DESCRIPTOR	1B-AP.	Algorithms & Programming
GRADE LEVEL EXPECTATION		Control
EXAMPLE	1B-AP-10.	Create programs that include sequences, events, loops, and conditionals. (P5.2)

Iowa Student Standards

Technology Education

Grade **4** - Adopted: **2018**

STRAND / COURSE		CSTA K-12 Computer Science Standards
ESSENTIAL CONCEPT AND/OR SKILL	CSTA.1B.	Level 1B (Ages 8-11)
DETAILED DESCRIPTOR	1B-AP.	Algorithms & Programming
GRADE LEVEL EXPECTATION		Variables
EXAMPLE	1B-AP-09.	Create programs that use variables to store and modify data. (P5.2)

STRAND / COURSE		CSTA K-12 Computer Science Standards
ESSENTIAL CONCEPT AND/OR SKILL	CSTA.1B.	Level 1B (Ages 8-11)
DETAILED DESCRIPTOR	1B-AP.	Algorithms & Programming
GRADE LEVEL EXPECTATION		Control
EXAMPLE	1B-AP-10.	Create programs that include sequences, events, loops, and conditionals. (P5.2)

Iowa Student Standards

Technology Education

Grade **5** - Adopted: **2018**

STRAND / COURSE		CSTA K-12 Computer Science Standards
ESSENTIAL CONCEPT AND/OR SKILL	CSTA.1B.	Level 1B (Ages 8-11)
DETAILED DESCRIPTOR	1B-AP.	Algorithms & Programming
GRADE LEVEL EXPECTATION		Variables

EXAMPLE	1B-AP-09.	Create programs that use variables to store and modify data. (P5.2)
STRAND / COURSE		CSTA K-12 Computer Science Standards
ESSENTIAL CONCEPT AND/OR SKILL	CSTA.1B.	Level 1B (Ages 8-11)
DETAILED DESCRIPTOR	1B-AP.	Algorithms & Programming
GRADE LEVEL EXPECTATION		Control
EXAMPLE	1B-AP-10.	Create programs that include sequences, events, loops, and conditionals. (P5.2)

Adventure with Lego: Lesson 2: Frozen Island

Iowa Student Standards

Mathematics

Grade **K** - Adopted: **2012**

STRAND / COURSE	IA.CC.K.MD.	Measurement and Data
ESSENTIAL CONCEPT AND/OR SKILL		Describe and compare measurable attributes.
DETAILED DESCRIPTOR	K.MD.1.	Describe measurable attributes of objects, such as length or weight. Describe several measurable attributes of a single object. (K.MD.1.) (DOK 2)

Iowa Student Standards

Mathematics

Grade **1** - Adopted: **2012**

STRAND / COURSE	IA.CC.1.OA.	Operations and Algebraic Thinking
ESSENTIAL CONCEPT AND/OR SKILL		Add and subtract within 20.
DETAILED DESCRIPTOR	1.OA.6.	Add and subtract within 20, demonstrating fluency for addition and subtraction within 10. Use strategies such as counting on; making ten (e.g., 8 + 6 = 8 + 2 + 4 = 10 + 4 = 14); decomposing a number leading to a ten (e.g., 13 – 4 = 13 – 3 – 1 = 10 – 1 = 9); using the relationship between addition and subtraction (e.g., knowing that 8 + 4 = 12, one knows 12 – 8 = 4); and creating equivalent but easier or known sums (e.g., adding 6 + 7 by creating the known equivalent 6 + 6 + 1 = 12 + 1 = 13). (1.OA.6.) (DOK 1,2)
STRAND / COURSE	IA.CC.1.NBT.	Number and Operations in Base Ten
ESSENTIAL CONCEPT AND/OR SKILL		Use place value understanding and properties of operations to add and subtract.
DETAILED DESCRIPTOR	1.NBT.4.	Add within 100, including adding a two-digit number and a one-digit number, and adding a two-digit number and a multiple of 10, using concrete models or drawings and strategies based on place value, properties of operations, and/or the relationship between addition and subtraction; relate the strategy to a written method and explain the reasoning used. Understand that in adding two-digit numbers, one adds tens and tens, ones and ones; and sometimes it is necessary to compose a ten. (1.NBT.4.) (DOK 1,2,3)

Iowa Student Standards

Mathematics

Grade **2** - Adopted: **2012**

STRAND / COURSE	IA.CC.2.NBT.	Number and Operations in Base Ten
ESSENTIAL CONCEPT AND/OR SKILL		Use place value understanding and properties of operations to add and subtract.
DETAILED	2.NBT.5.	Fluently add and subtract within 100 using strategies based on place value,

DESCRIPTOR		properties of operations, and/or the relationship between addition and subtraction. (2.NBT.5.) (DOK 1,2)
DETAILED DESCRIPTOR	2.NBT.6.	Add up to four two-digit numbers using strategies based on place value and properties of operations. (2.NBT.6.) (DOK 2)
DETAILED DESCRIPTOR	2.NBT.7.	Add and subtract within 1000, using concrete models or drawings and strategies based on place value, properties of operations, and/or the relationship between addition and subtraction; relate the strategy to a written method. Understand that in adding or subtracting three-digit numbers, one adds or subtracts hundreds and hundreds, tens and tens, ones and ones; and sometimes it is necessary to compose or decompose tens or hundreds. (2.NBT.7.) (DOK 2)
DETAILED DESCRIPTOR	2.NBT.9.	Explain why addition and subtraction strategies work, using place value and the properties of operations. (2.NBT.9.) (DOK 3)
STRAND / COURSE	IA.CC.2. MD.	Measurement and Data
ESSENTIAL CONCEPT AND/OR SKILL		Measure and estimate lengths in standard units.
DETAILED DESCRIPTOR	2.MD.1.	Measure the length of an object by selecting and using appropriate tools such as rulers, yardsticks, meter sticks, and measuring tapes. (2.MD.1.) (DOK 1)
STRAND / COURSE	IA.CC.2. MD.	Measurement and Data
ESSENTIAL CONCEPT AND/OR SKILL		Represent and interpret data.
DETAILED DESCRIPTOR	2.MD.9.	Generate measurement data by measuring lengths of several objects to the nearest whole unit, or by making repeated measurements of the same object. Show the measurements by making a line plot, where the horizontal scale is marked off in whole-number units. (2.MD.9.) (DOK 2)

Iowa Student Standards

Mathematics

Grade **3** - Adopted: **2012**

STRAND / COURSE	IA.CC.3. OA.	Operations and Algebraic Thinking
ESSENTIAL CONCEPT AND/OR SKILL		Multiply and divide within 100.
DETAILED DESCRIPTOR	3.OA.7.	Fluently multiply and divide within 100, using strategies such as the relationship between multiplication and division (e.g., knowing that 8 × 5 = 40, one knows 40 ÷ 5 = 8) or properties of operations. By the end of Grade 3, know from memory all products of two one-digit numbers. (3.OA.7.) (DOK 1,2)
STRAND / COURSE	IA.CC.3. NBT.	Number and Operations in Base Ten
ESSENTIAL CONCEPT AND/OR SKILL		Use place value understanding and properties of operations to perform multi-digit arithmetic.
DETAILED DESCRIPTOR	3.NBT.2.	Fluently add and subtract within 1000 using strategies and algorithms based on place value, properties of operations, and/or the relationship between addition and subtraction. (3.NBT.2.) (DOK 1,2)

Iowa Student Standards

Mathematics

Grade **4** - Adopted: **2012**

STRAND / COURSE	IA.CC.4. NBT.	Number and Operations in Base Ten
ESSENTIAL CONCEPT AND/OR SKILL		Use place value understanding and properties of operations to perform multi-digit arithmetic.
DETAILED DESCRIPTOR	4.NBT.4.	Fluently add and subtract multi-digit whole numbers using the standard algorithm. (4.NBT.4.) (DOK 1)
DETAILED DESCRIPTOR	4.NBT.6.	Find whole-number quotients and remainders with up to four-digit dividends and one-digit divisors, using strategies based on place value, the properties of operations, and/or the relationship between multiplication and division. Illustrate and explain the calculation by using equations, rectangular arrays, and/or area models. (4.NBT.6.) (DOK 1,2)

STRAND / COURSE	IA.CC.4.MD.	Measurement and Data
ESSENTIAL CONCEPT AND/OR SKILL		Geometric measurement: understand concepts of angle and measure angles.
DETAILED DESCRIPTOR	4.MD.5.	Recognize angles as geometric shapes that are formed wherever two rays share a common endpoint, and understand concepts of angle measurement:
GRADE LEVEL EXPECTATION	4.MD.5.b.	An angle that turns through n one-degree angles is said to have an angle measure of n degrees. (4.MD.5.) (DOK 1)
STRAND / COURSE	IA.CC.4.MD.	Measurement and Data
ESSENTIAL CONCEPT AND/OR SKILL		Geometric measurement: understand concepts of angle and measure angles.
DETAILED DESCRIPTOR	4.MD.6.	Measure angles in whole-number degrees using a protractor. Sketch angles of specified measure. (4.MD.6.) (DOK 1)

Iowa Student Standards

Mathematics

Grade **5** - Adopted: **2012**

STRAND / COURSE	IA.CC.5.NBT.	Number and Operations in Base Ten
ESSENTIAL CONCEPT AND/OR SKILL		Perform operations with multi-digit whole numbers and with decimals to hundredths.
DETAILED DESCRIPTOR	5.NBT.6.	Find whole-number quotients of whole numbers with up to four-digit dividends and two-digit divisors, using strategies based on place value, the properties of operations, and/or the relationship between multiplication and division. Illustrate and explain the calculation by using equations, rectangular arrays, and/or area models. (5.NBT.6.) (DOK 1,2)

Iowa Student Standards

Science

Grade **K** - Adopted: **2015**

STRAND / COURSE	IA.K-PS2.	Motion and Stability: Forces and interactions
ESSENTIAL CONCEPT AND/OR SKILL		Students who demonstrate understanding can:
DETAILED DESCRIPTOR	K-PS2-1.	Plan and conduct an investigation to compare the effects of different strengths or different directions of pushes and pulls on the motion of an object.
STRAND / COURSE	IA.K-2-ETS1.	Engineering Design
ESSENTIAL CONCEPT AND/OR SKILL		Students who demonstrate understanding can:
DETAILED DESCRIPTOR	K-2-ETS1-1.	Ask questions, make observations, and gather information about a situation people want to change to define a simple problem that can be solved through the development of a new or improved object or tool.
DETAILED DESCRIPTOR	K-2-ETS1-2.	Develop a simple sketch, drawing, or physical model to illustrate how the shape of an object helps it function as needed to solve a given problem.

Iowa Student Standards

Science

Grade **1** - Adopted: **2015**

STRAND / COURSE	IA.K-2-ETS1.	Engineering Design
ESSENTIAL CONCEPT AND/OR SKILL		Students who demonstrate understanding can:
DETAILED DESCRIPTOR	K-2-ETS1-1.	Ask questions, make observations, and gather information about a situation people want to change to define a simple problem that can be solved through the development of a new or improved object or tool.
DETAILED	K-2-	Develop a simple sketch, drawing, or physical model to illustrate how the shape of an

DESCRIPTOR	ETS1-2.	object helps it function as needed to solve a given problem.

Iowa Student Standards

Science

Grade **2** - Adopted: **2015**

STRAND / COURSE	IA.2-PS1.	Matter and its Interactions
ESSENTIAL CONCEPT AND/OR SKILL		Students who demonstrate understanding can:
DETAILED DESCRIPTOR	2-PS1-2.	Analyze data obtained from testing different materials to determine which materials have the properties that are best suited for an intended purpose.
DETAILED DESCRIPTOR	2-PS1-3.	Make observations to construct an evidence-based account of how an object made of a small set of pieces can be disassembled and made into a new object.
STRAND / COURSE	IA.K-2-ETS1.	Engineering Design
ESSENTIAL CONCEPT AND/OR SKILL		Students who demonstrate understanding can:
DETAILED DESCRIPTOR	K-2-ETS1-1.	Ask questions, make observations, and gather information about a situation people want to change to define a simple problem that can be solved through the development of a new or improved object or tool.
DETAILED DESCRIPTOR	K-2-ETS1-2.	Develop a simple sketch, drawing, or physical model to illustrate how the shape of an object helps it function as needed to solve a given problem.

Iowa Student Standards

Science

Grade **3** - Adopted: **2015**

STRAND / COURSE	IA.3-PS2.	Motion and Stability: Forces and Interactions
ESSENTIAL CONCEPT AND/OR SKILL		Students who demonstrate understanding can:
DETAILED DESCRIPTOR	3-PS2-2.	Make observations and/or measurements of an object's motion to provide evidence that a pattern can be used to predict future motion.
STRAND / COURSE	IA.3-5-ETS1.	Engineering Design
ESSENTIAL CONCEPT AND/OR SKILL		Students who demonstrate understanding can:
DETAILED DESCRIPTOR	3-5-ETS1-1.	Define a simple design problem reflecting a need or a want that includes specified criteria for success and constraints on materials, time, or cost.
DETAILED DESCRIPTOR	3-5-ETS1-2.	Generate and compare multiple possible solutions to a problem based on how well each is likely to meet the criteria and constraints of the problem.
DETAILED DESCRIPTOR	3-5-ETS1-3.	Plan and carry out fair tests in which variables are controlled and failure points are considered to identify aspects of a model or prototype that can be improved.

Iowa Student Standards

Science

Grade **4** - Adopted: **2015**

STRAND / COURSE	IA.3-5-ETS1.	Engineering Design
ESSENTIAL CONCEPT AND/OR SKILL		Students who demonstrate understanding can:
DETAILED DESCRIPTOR	3-5-ETS1-1.	Define a simple design problem reflecting a need or a want that includes specified criteria for success and constraints on materials, time, or cost.
DETAILED DESCRIPTOR	3-5-ETS1-2.	Generate and compare multiple possible solutions to a problem based on how well each is likely to meet the criteria and constraints of the problem.
DETAILED DESCRIPTOR	3-5-ETS1-3.	Plan and carry out fair tests in which variables are controlled and failure points are considered to identify aspects of a model or prototype that can be improved.

Iowa Student Standards

Science

Grade **5** - Adopted: **2015**

STRAND / COURSE	IA.3-5-ETS1.	Engineering Design
ESSENTIAL CONCEPT AND/OR SKILL		Students who demonstrate understanding can:
DETAILED DESCRIPTOR	3-5-ETS1-1.	Define a simple design problem reflecting a need or a want that includes specified criteria for success and constraints on materials, time, or cost.
DETAILED DESCRIPTOR	3-5-ETS1-2.	Generate and compare multiple possible solutions to a problem based on how well each is likely to meet the criteria and constraints of the problem.
DETAILED DESCRIPTOR	3-5-ETS1-3.	Plan and carry out fair tests in which variables are controlled and failure points are considered to identify aspects of a model or prototype that can be improved.

Iowa Student Standards

Technology Education

Grade **K** - Adopted: **2018**

STRAND / COURSE		CSTA K-12 Computer Science Standards
ESSENTIAL CONCEPT AND/OR SKILL	CSTA.1A.	Level 1A (Ages 5-7)
DETAILED DESCRIPTOR	1A-CS.	Computing Systems
GRADE LEVEL EXPECTATION		Devices
EXAMPLE	1A-CS-01.	Select and operate appropriate software to perform a variety of tasks, and recognize that users have different needs and preferences for the technology they use. (P1.1)
STRAND / COURSE		CSTA K-12 Computer Science Standards
ESSENTIAL CONCEPT AND/OR SKILL	CSTA.1A.	Level 1A (Ages 5-7)
DETAILED DESCRIPTOR	1A-AP.	Algorithms & Programming
GRADE LEVEL EXPECTATION		Program Development
EXAMPLE	1A-AP-12.	Develop plans that describe a program's sequence of events, goals, and expected outcomes. (P5.1, P7.2)
EXAMPLE	1A-AP-15.	Using correct terminology, describe steps taken and choices made during the iterative process of program development. (P7.2)

Iowa Student Standards

Technology Education

Grade **1** - Adopted: **2018**

STRAND / COURSE		CSTA K-12 Computer Science Standards
ESSENTIAL CONCEPT AND/OR SKILL	CSTA.1A.	Level 1A (Ages 5-7)
DETAILED DESCRIPTOR	1A-CS.	Computing Systems
GRADE LEVEL EXPECTATION		Devices
EXAMPLE	1A-CS-01.	Select and operate appropriate software to perform a variety of tasks, and recognize that users have different needs and preferences for the technology they use. (P1.1)
STRAND / COURSE		CSTA K-12 Computer Science Standards
ESSENTIAL CONCEPT AND/OR SKILL	CSTA.1A.	Level 1A (Ages 5-7)
DETAILED	1A-AP.	Algorithms & Programming

DESCRIPTOR		
GRADE LEVEL EXPECTATION		Program Development
EXAMPLE	1A-AP-12.	Develop plans that describe a program’s sequence of events, goals, and expected outcomes. (P5.1, P7.2)
EXAMPLE	1A-AP-15.	Using correct terminology, describe steps taken and choices made during the iterative process of program development. (P7.2)

Iowa Student Standards

Technology Education

Grade **2** - Adopted: **2018**

STRAND / COURSE		CSTA K-12 Computer Science Standards
ESSENTIAL CONCEPT AND/OR SKILL	CSTA.1A.	Level 1A (Ages 5-7)
DETAILED DESCRIPTOR	1A-CS.	Computing Systems
GRADE LEVEL EXPECTATION		Devices
EXAMPLE	1A-CS-01.	Select and operate appropriate software to perform a variety of tasks, and recognize that users have different needs and preferences for the technology they use. (P1.1)

STRAND / COURSE		CSTA K-12 Computer Science Standards
ESSENTIAL CONCEPT AND/OR SKILL	CSTA.1A.	Level 1A (Ages 5-7)
DETAILED DESCRIPTOR	1A-AP.	Algorithms & Programming
GRADE LEVEL EXPECTATION		Program Development
EXAMPLE	1A-AP-12.	Develop plans that describe a program’s sequence of events, goals, and expected outcomes. (P5.1, P7.2)
EXAMPLE	1A-AP-15.	Using correct terminology, describe steps taken and choices made during the iterative process of program development. (P7.2)

Iowa Student Standards

Technology Education

Grade **3** - Adopted: **2018**

STRAND / COURSE		CSTA K-12 Computer Science Standards
ESSENTIAL CONCEPT AND/OR SKILL	CSTA.1B.	Level 1B (Ages 8-11)
DETAILED DESCRIPTOR	1B-AP.	Algorithms & Programming
GRADE LEVEL EXPECTATION		Variables
EXAMPLE	1B-AP-09.	Create programs that use variables to store and modify data. (P5.2)

STRAND / COURSE		CSTA K-12 Computer Science Standards
ESSENTIAL CONCEPT AND/OR SKILL	CSTA.1B.	Level 1B (Ages 8-11)
DETAILED DESCRIPTOR	1B-AP.	Algorithms & Programming
GRADE LEVEL EXPECTATION		Control
EXAMPLE	1B-AP-10.	Create programs that include sequences, events, loops, and conditionals. (P5.2)

STRAND / COURSE		CSTA K-12 Computer Science Standards

ESSENTIAL CONCEPT AND/OR SKILL	CSTA.1B.	Level 1B (Ages 8-11)
DETAILED DESCRIPTOR	1B-AP.	Algorithms & Programming
GRADE LEVEL EXPECTATION		Program Development
EXAMPLE	1B-AP-17.	Describe choices made during program development using code comments, presentations, and demonstrations. (P7.2)

Iowa Student Standards

Technology Education

Grade **4** - Adopted: **2018**

STRAND / COURSE		CSTA K-12 Computer Science Standards
ESSENTIAL CONCEPT AND/OR SKILL	CSTA.1B.	Level 1B (Ages 8-11)
DETAILED DESCRIPTOR	1B-AP.	Algorithms & Programming
GRADE LEVEL EXPECTATION		Variables
EXAMPLE	1B-AP-09.	Create programs that use variables to store and modify data. (P5.2)
STRAND / COURSE		CSTA K-12 Computer Science Standards
ESSENTIAL CONCEPT AND/OR SKILL	CSTA.1B.	Level 1B (Ages 8-11)
DETAILED DESCRIPTOR	1B-AP.	Algorithms & Programming
GRADE LEVEL EXPECTATION		Control
EXAMPLE	1B-AP-10.	Create programs that include sequences, events, loops, and conditionals. (P5.2)
STRAND / COURSE		CSTA K-12 Computer Science Standards
ESSENTIAL CONCEPT AND/OR SKILL	CSTA.1B.	Level 1B (Ages 8-11)
DETAILED DESCRIPTOR	1B-AP.	Algorithms & Programming
GRADE LEVEL EXPECTATION		Program Development
EXAMPLE	1B-AP-17.	Describe choices made during program development using code comments, presentations, and demonstrations. (P7.2)

Iowa Student Standards

Technology Education

Grade **5** - Adopted: **2018**

STRAND / COURSE		CSTA K-12 Computer Science Standards
ESSENTIAL CONCEPT AND/OR SKILL	CSTA.1B.	Level 1B (Ages 8-11)
DETAILED DESCRIPTOR	1B-AP.	Algorithms & Programming
GRADE LEVEL EXPECTATION		Variables
EXAMPLE	1B-AP-09.	Create programs that use variables to store and modify data. (P5.2)
STRAND / COURSE		CSTA K-12 Computer Science Standards

ESSENTIAL CONCEPT AND/OR SKILL	CSTA.1B.	Level 1B (Ages 8-11)
DETAILED DESCRIPTOR	1B-AP.	Algorithms & Programming
GRADE LEVEL EXPECTATION		Control
EXAMPLE	1B-AP-10.	Create programs that include sequences, events, loops, and conditionals. (P5.2)
STRAND / COURSE		CSTA K-12 Computer Science Standards
ESSENTIAL CONCEPT AND/OR SKILL	CSTA.1B.	Level 1B (Ages 8-11)
DETAILED DESCRIPTOR	1B-AP.	Algorithms & Programming
GRADE LEVEL EXPECTATION		Program Development
EXAMPLE	1B-AP-17.	Describe choices made during program development using code comments, presentations, and demonstrations. (P7.2)

Adventure with Lego: Lesson 3: The Lost City

Iowa Student Standards

Mathematics

Grade **K** - Adopted: **2012**

STRAND / COURSE	IA.CC.K.MD.	Measurement and Data
ESSENTIAL CONCEPT AND/OR SKILL		Describe and compare measurable attributes.
DETAILED DESCRIPTOR	K.MD.1.	Describe measurable attributes of objects, such as length or weight. Describe several measurable attributes of a single object. (K.MD.1.) (DOK 2)

Iowa Student Standards

Mathematics

Grade **2** - Adopted: **2012**

STRAND / COURSE	IA.CC.2.MD.	Measurement and Data
ESSENTIAL CONCEPT AND/OR SKILL		Measure and estimate lengths in standard units.
DETAILED DESCRIPTOR	2.MD.1.	Measure the length of an object by selecting and using appropriate tools such as rulers, yardsticks, meter sticks, and measuring tapes. (2.MD.1.) (DOK 1)
STRAND / COURSE	IA.CC.2.MD.	Measurement and Data
ESSENTIAL CONCEPT AND/OR SKILL		Represent and interpret data.
DETAILED DESCRIPTOR	2.MD.9.	Generate measurement data by measuring lengths of several objects to the nearest whole unit, or by making repeated measurements of the same object. Show the measurements by making a line plot, where the horizontal scale is marked off in whole-number units. (2.MD.9.) (DOK 2)

Iowa Student Standards

Mathematics

Grade **4** - Adopted: **2012**

STRAND / COURSE	IA.CC.4.MD.	Measurement and Data
ESSENTIAL CONCEPT AND/OR SKILL		Geometric measurement: understand concepts of angle and measure angles.
DETAILED	4.MD.5.	Recognize angles as geometric shapes that are formed wherever two rays share a

DESCRIPTOR		common endpoint, and understand concepts of angle measurement:
GRADE LEVEL EXPECTATION	4.MD.5.b.	An angle that turns through n one-degree angles is said to have an angle measure of n degrees. (4.MD.5.) (DOK 1)
STRAND / COURSE	IA.CC.4.MD.	Measurement and Data
ESSENTIAL CONCEPT AND/OR SKILL		Geometric measurement: understand concepts of angle and measure angles.
DETAILED DESCRIPTOR	4.MD.6.	Measure angles in whole-number degrees using a protractor. Sketch angles of specified measure. (4.MD.6.) (DOK 1)

Iowa Student Standards

Science

Grade **K** - Adopted: **2015**

STRAND / COURSE	IA.K-PS2.	Motion and Stability: Forces and interactions
ESSENTIAL CONCEPT AND/OR SKILL		Students who demonstrate understanding can:
DETAILED DESCRIPTOR	K-PS2-1.	Plan and conduct an investigation to compare the effects of different strengths or different directions of pushes and pulls on the motion of an object.
STRAND / COURSE	IA.K-2-ETS1.	Engineering Design
ESSENTIAL CONCEPT AND/OR SKILL		Students who demonstrate understanding can:
DETAILED DESCRIPTOR	K-2-ETS1-1.	Ask questions, make observations, and gather information about a situation people want to change to define a simple problem that can be solved through the development of a new or improved object or tool.
DETAILED DESCRIPTOR	K-2-ETS1-2.	Develop a simple sketch, drawing, or physical model to illustrate how the shape of an object helps it function as needed to solve a given problem.

Iowa Student Standards

Science

Grade **1** - Adopted: **2015**

STRAND / COURSE	IA.K-2-ETS1.	Engineering Design
ESSENTIAL CONCEPT AND/OR SKILL		Students who demonstrate understanding can:
DETAILED DESCRIPTOR	K-2-ETS1-1.	Ask questions, make observations, and gather information about a situation people want to change to define a simple problem that can be solved through the development of a new or improved object or tool.
DETAILED DESCRIPTOR	K-2-ETS1-2.	Develop a simple sketch, drawing, or physical model to illustrate how the shape of an object helps it function as needed to solve a given problem.

Iowa Student Standards

Science

Grade **2** - Adopted: **2015**

STRAND / COURSE	IA.2-PS1.	Matter and its Interactions
ESSENTIAL CONCEPT AND/OR SKILL		Students who demonstrate understanding can:
DETAILED DESCRIPTOR	2-PS1-2.	Analyze data obtained from testing different materials to determine which materials have the properties that are best suited for an intended purpose.
DETAILED DESCRIPTOR	2-PS1-3.	Make observations to construct an evidence-based account of how an object made of a small set of pieces can be disassembled and made into a new object.
STRAND / COURSE	IA.K-2-ETS1.	Engineering Design
ESSENTIAL CONCEPT AND/OR SKILL		Students who demonstrate understanding can:

DETAILED DESCRIPTOR	K-2-ETS1-1.	Ask questions, make observations, and gather information about a situation people want to change to define a simple problem that can be solved through the development of a new or improved object or tool.
DETAILED DESCRIPTOR	K-2-ETS1-2.	Develop a simple sketch, drawing, or physical model to illustrate how the shape of an object helps it function as needed to solve a given problem.

Iowa Student Standards

Science

Grade **3** - Adopted: **2015**

STRAND / COURSE	IA.3-PS2.	Motion and Stability: Forces and Interactions
ESSENTIAL CONCEPT AND/OR SKILL		Students who demonstrate understanding can:
DETAILED DESCRIPTOR	3-PS2-2.	Make observations and/or measurements of an object's motion to provide evidence that a pattern can be used to predict future motion.
STRAND / COURSE	IA.3-5-ETS1.	Engineering Design
ESSENTIAL CONCEPT AND/OR SKILL		Students who demonstrate understanding can:
DETAILED DESCRIPTOR	3-5-ETS1-1.	Define a simple design problem reflecting a need or a want that includes specified criteria for success and constraints on materials, time, or cost.
DETAILED DESCRIPTOR	3-5-ETS1-2.	Generate and compare multiple possible solutions to a problem based on how well each is likely to meet the criteria and constraints of the problem.
DETAILED DESCRIPTOR	3-5-ETS1-3.	Plan and carry out fair tests in which variables are controlled and failure points are considered to identify aspects of a model or prototype that can be improved.

Iowa Student Standards

Science

Grade **4** - Adopted: **2015**

STRAND / COURSE	IA.3-5-ETS1.	Engineering Design
ESSENTIAL CONCEPT AND/OR SKILL		Students who demonstrate understanding can:
DETAILED DESCRIPTOR	3-5-ETS1-1.	Define a simple design problem reflecting a need or a want that includes specified criteria for success and constraints on materials, time, or cost.
DETAILED DESCRIPTOR	3-5-ETS1-2.	Generate and compare multiple possible solutions to a problem based on how well each is likely to meet the criteria and constraints of the problem.
DETAILED DESCRIPTOR	3-5-ETS1-3.	Plan and carry out fair tests in which variables are controlled and failure points are considered to identify aspects of a model or prototype that can be improved.

Iowa Student Standards

Science

Grade **5** - Adopted: **2015**

STRAND / COURSE	IA.3-5-ETS1.	Engineering Design
ESSENTIAL CONCEPT AND/OR SKILL		Students who demonstrate understanding can:
DETAILED DESCRIPTOR	3-5-ETS1-1.	Define a simple design problem reflecting a need or a want that includes specified criteria for success and constraints on materials, time, or cost.
DETAILED DESCRIPTOR	3-5-ETS1-2.	Generate and compare multiple possible solutions to a problem based on how well each is likely to meet the criteria and constraints of the problem.
DETAILED DESCRIPTOR	3-5-ETS1-3.	Plan and carry out fair tests in which variables are controlled and failure points are considered to identify aspects of a model or prototype that can be improved.

Iowa Student Standards

Technology Education

Grade **K** - Adopted: **2018**

STRAND /		CSTA K-12 Computer Science Standards

COURSE		
ESSENTIAL CONCEPT AND/OR SKILL	CSTA.1A.	Level 1A (Ages 5-7)
DETAILED DESCRIPTOR	1A-CS.	Computing Systems
GRADE LEVEL EXPECTATION		Devices
EXAMPLE	1A-CS-01.	Select and operate appropriate software to perform a variety of tasks, and recognize that users have different needs and preferences for the technology they use. (P1.1)

STRAND / COURSE		CSTA K-12 Computer Science Standards
ESSENTIAL CONCEPT AND/OR SKILL	CSTA.1A.	Level 1A (Ages 5-7)
DETAILED DESCRIPTOR	1A-AP.	Algorithms & Programming
GRADE LEVEL EXPECTATION		Control
EXAMPLE	1A-AP-10.	Develop programs with sequences and simple loops, to express ideas or address a problem. (P5.2)

STRAND / COURSE		CSTA K-12 Computer Science Standards
ESSENTIAL CONCEPT AND/OR SKILL	CSTA.1A.	Level 1A (Ages 5-7)
DETAILED DESCRIPTOR	1A-AP.	Algorithms & Programming
GRADE LEVEL EXPECTATION		Program Development
EXAMPLE	1A-AP-12.	Develop plans that describe a program's sequence of events, goals, and expected outcomes. (P5.1, P7.2)
EXAMPLE	1A-AP-15.	Using correct terminology, describe steps taken and choices made during the iterative process of program development. (P7.2)

Iowa Student Standards

Technology Education

Grade **1** - Adopted: **2018**

STRAND / COURSE		CSTA K-12 Computer Science Standards
ESSENTIAL CONCEPT AND/OR SKILL	CSTA.1A.	Level 1A (Ages 5-7)
DETAILED DESCRIPTOR	1A-CS.	Computing Systems
GRADE LEVEL EXPECTATION		Devices
EXAMPLE	1A-CS-01.	Select and operate appropriate software to perform a variety of tasks, and recognize that users have different needs and preferences for the technology they use. (P1.1)

STRAND / COURSE		CSTA K-12 Computer Science Standards
ESSENTIAL CONCEPT AND/OR SKILL	CSTA.1A.	Level 1A (Ages 5-7)
DETAILED DESCRIPTOR	1A-AP.	Algorithms & Programming
GRADE LEVEL EXPECTATION		Control
EXAMPLE	1A-AP-10.	Develop programs with sequences and simple loops, to express ideas or address a problem. (P5.2)

STRAND / COURSE		CSTA K-12 Computer Science Standards
ESSENTIAL CONCEPT	CSTA.1A.	Level 1A (Ages 5-7)

AND/OR SKILL		
DETAILED DESCRIPTOR	1A-AP.	Algorithms & Programming
GRADE LEVEL EXPECTATION		Program Development
EXAMPLE	1A-AP-12.	Develop plans that describe a program's sequence of events, goals, and expected outcomes. (P5.1, P7.2)
EXAMPLE	1A-AP-15.	Using correct terminology, describe steps taken and choices made during the iterative process of program development. (P7.2)

Iowa Student Standards

Technology Education

Grade **2** - Adopted: **2018**

STRAND / COURSE		CSTA K-12 Computer Science Standards
ESSENTIAL CONCEPT AND/OR SKILL	CSTA.1A.	Level 1A (Ages 5-7)
DETAILED DESCRIPTOR	1A-CS.	Computing Systems
GRADE LEVEL EXPECTATION		Devices
EXAMPLE	1A-CS-01.	Select and operate appropriate software to perform a variety of tasks, and recognize that users have different needs and preferences for the technology they use. (P1.1)
STRAND / COURSE		CSTA K-12 Computer Science Standards
ESSENTIAL CONCEPT AND/OR SKILL	CSTA.1A.	Level 1A (Ages 5-7)
DETAILED DESCRIPTOR	1A-AP.	Algorithms & Programming
GRADE LEVEL EXPECTATION		Control
EXAMPLE	1A-AP-10.	Develop programs with sequences and simple loops, to express ideas or address a problem. (P5.2)
STRAND / COURSE		CSTA K-12 Computer Science Standards
ESSENTIAL CONCEPT AND/OR SKILL	CSTA.1A.	Level 1A (Ages 5-7)
DETAILED DESCRIPTOR	1A-AP.	Algorithms & Programming
GRADE LEVEL EXPECTATION		Program Development
EXAMPLE	1A-AP-12.	Develop plans that describe a program's sequence of events, goals, and expected outcomes. (P5.1, P7.2)
EXAMPLE	1A-AP-15.	Using correct terminology, describe steps taken and choices made during the iterative process of program development. (P7.2)

Iowa Student Standards

Technology Education

Grade **3** - Adopted: **2018**

STRAND / COURSE		CSTA K-12 Computer Science Standards
ESSENTIAL CONCEPT AND/OR SKILL	CSTA.1B.	Level 1B (Ages 8-11)
DETAILED DESCRIPTOR	1B-AP.	Algorithms & Programming
GRADE LEVEL EXPECTATION		Variables
EXAMPLE	1B-AP-09.	Create programs that use variables to store and modify data. (P5.2)

STRAND / COURSE		CSTA K-12 Computer Science Standards
ESSENTIAL CONCEPT AND/OR SKILL	CSTA.1B.	Level 1B (Ages 8-11)
DETAILED DESCRIPTOR	1B-AP.	Algorithms & Programming
GRADE LEVEL EXPECTATION		Control
EXAMPLE	1B-AP-10.	Create programs that include sequences, events, loops, and conditionals. (P5.2)

STRAND / COURSE		CSTA K-12 Computer Science Standards
ESSENTIAL CONCEPT AND/OR SKILL	CSTA.1B.	Level 1B (Ages 8-11)
DETAILED DESCRIPTOR	1B-AP.	Algorithms & Programming
GRADE LEVEL EXPECTATION		Program Development
EXAMPLE	1B-AP-13.	Use an iterative process to plan the development of a program by including others'' perspectives and considering user preferences. (P1.1, P5.1)

Iowa Student Standards

Technology Education

Grade **4** - Adopted: **2018**

STRAND / COURSE		CSTA K-12 Computer Science Standards
ESSENTIAL CONCEPT AND/OR SKILL	CSTA.1B.	Level 1B (Ages 8-11)
DETAILED DESCRIPTOR	1B-AP.	Algorithms & Programming
GRADE LEVEL EXPECTATION		Variables
EXAMPLE	1B-AP-09.	Create programs that use variables to store and modify data. (P5.2)

STRAND / COURSE		CSTA K-12 Computer Science Standards
ESSENTIAL CONCEPT AND/OR SKILL	CSTA.1B.	Level 1B (Ages 8-11)
DETAILED DESCRIPTOR	1B-AP.	Algorithms & Programming
GRADE LEVEL EXPECTATION		Control
EXAMPLE	1B-AP-10.	Create programs that include sequences, events, loops, and conditionals. (P5.2)

STRAND / COURSE		CSTA K-12 Computer Science Standards
ESSENTIAL CONCEPT AND/OR SKILL	CSTA.1B.	Level 1B (Ages 8-11)
DETAILED DESCRIPTOR	1B-AP.	Algorithms & Programming
GRADE LEVEL EXPECTATION		Program Development
EXAMPLE	1B-AP-13.	Use an iterative process to plan the development of a program by including others'' perspectives and considering user preferences. (P1.1, P5.1)

Iowa Student Standards

Technology Education

Grade **5** - Adopted: **2018**

STRAND / COURSE		CSTA K-12 Computer Science Standards
ESSENTIAL CONCEPT AND/OR SKILL	CSTA.1B.	Level 1B (Ages 8-11)
DETAILED DESCRIPTOR	1B-AP.	Algorithms & Programming
GRADE LEVEL EXPECTATION		Variables
EXAMPLE	1B-AP-09.	Create programs that use variables to store and modify data. (P5.2)

STRAND / COURSE		CSTA K-12 Computer Science Standards
ESSENTIAL CONCEPT AND/OR SKILL	CSTA.1B.	Level 1B (Ages 8-11)
DETAILED DESCRIPTOR	1B-AP.	Algorithms & Programming
GRADE LEVEL EXPECTATION		Control
EXAMPLE	1B-AP-10.	Create programs that include sequences, events, loops, and conditionals. (P5.2)

STRAND / COURSE		CSTA K-12 Computer Science Standards
ESSENTIAL CONCEPT AND/OR SKILL	CSTA.1B.	Level 1B (Ages 8-11)
DETAILED DESCRIPTOR	1B-AP.	Algorithms & Programming
GRADE LEVEL EXPECTATION		Program Development
EXAMPLE	1B-AP-13.	Use an iterative process to plan the development of a program by including others'' perspectives and considering user preferences. (P1.1, P5.1)

Adventure with Lego: Lesson 4: Crystal Crater

Iowa Student Standards

Mathematics

Grade **K** - Adopted: **2012**

STRAND / COURSE	IA.CC.K.MD.	Measurement and Data
ESSENTIAL CONCEPT AND/OR SKILL		Describe and compare measurable attributes.
DETAILED DESCRIPTOR	K.MD.1.	Describe measurable attributes of objects, such as length or weight. Describe several measurable attributes of a single object. (K.MD.1.) (DOK 2)

Iowa Student Standards

Mathematics

Grade **1** - Adopted: **2012**

STRAND / COURSE	IA.CC.1.G.	Geometry
ESSENTIAL CONCEPT AND/OR SKILL		Reason with shapes and their attributes.
DETAILED DESCRIPTOR	1.G.1.	Distinguish between defining attributes (e.g., triangles are closed and three-sided) versus non-defining attributes (e.g., color, orientation, overall size); build and draw shapes to possess defining attributes. (1.G.1.) (DOK 2)

Iowa Student Standards

Mathematics

Grade **2** - Adopted: **2012**

STRAND / COURSE	IA.CC.2.MD.	Measurement and Data
ESSENTIAL CONCEPT AND/OR SKILL		Measure and estimate lengths in standard units.
DETAILED DESCRIPTOR	2.MD.1.	Measure the length of an object by selecting and using appropriate tools such as rulers, yardsticks, meter sticks, and measuring tapes. (2.MD.1.) (DOK 1)
STRAND / COURSE	IA.CC.2.MD.	Measurement and Data
ESSENTIAL CONCEPT AND/OR SKILL		Represent and interpret data.
DETAILED DESCRIPTOR	2.MD.9.	Generate measurement data by measuring lengths of several objects to the nearest whole unit, or by making repeated measurements of the same object. Show the measurements by making a line plot, where the horizontal scale is marked off in whole-number units. (2.MD.9.) (DOK 2)

Iowa Student Standards

Mathematics

Grade **4** - Adopted: **2012**

STRAND / COURSE	IA.CC.4.MD.	Measurement and Data
ESSENTIAL CONCEPT AND/OR SKILL		Geometric measurement: understand concepts of angle and measure angles.
DETAILED DESCRIPTOR	4.MD.5.	Recognize angles as geometric shapes that are formed wherever two rays share a common endpoint, and understand concepts of angle measurement:
GRADE LEVEL EXPECTATION	4.MD.5.a.	An angle is measured with reference to a circle with its center at the common endpoint of the rays, by considering the fraction of the circular arc between the points where the two rays intersect the circle. An angle that turns through 1/360 of a circle is called a "one-degree angle," and can be used to measure angles. (4.MD.5.) (DOK 1)
GRADE LEVEL EXPECTATION	4.MD.5.b.	An angle that turns through n one-degree angles is said to have an angle measure of n degrees. (4.MD.5.) (DOK 1)
STRAND / COURSE	IA.CC.4.MD.	Measurement and Data
ESSENTIAL CONCEPT AND/OR SKILL		Geometric measurement: understand concepts of angle and measure angles.
DETAILED DESCRIPTOR	4.MD.6.	Measure angles in whole-number degrees using a protractor. Sketch angles of specified measure. (4.MD.6.) (DOK 1)
DETAILED DESCRIPTOR	4.MD.7.	Recognize angle measure as additive. When an angle is decomposed into non-overlapping parts, the angle measure of the whole is the sum of the angle measures of the parts. Solve addition and subtraction problems to find unknown angles on a diagram in real world and mathematical problems, e.g., by using an equation with a symbol for the unknown angle measure. (4.MD.7.) (DOK 1,2)
STRAND / COURSE	IA.CC.4.G.	Geometry
ESSENTIAL CONCEPT AND/OR SKILL		Draw and identify lines and angles, and classify shapes by properties of their lines and angles.
DETAILED DESCRIPTOR	4.G.1.	Draw points, lines, line segments, rays, angles (right, acute, obtuse), and perpendicular and parallel lines. Identify these in two-dimensional figures. (4.G.1.) (DOK 1)

Iowa Student Standards

Science

Grade **K** - Adopted: **2015**

STRAND / COURSE	IA.K-PS2.	Motion and Stability: Forces and interactions
ESSENTIAL CONCEPT AND/OR SKILL		Students who demonstrate understanding can:
DETAILED DESCRIPTOR	K-PS2-1.	Plan and conduct an investigation to compare the effects of different strengths or different directions of pushes and pulls on the motion of an object.

STRAND / COURSE	IA.K-2-ETS1.	Engineering Design
ESSENTIAL CONCEPT AND/OR SKILL		Students who demonstrate understanding can:
DETAILED DESCRIPTOR	K-2-ETS1-1.	Ask questions, make observations, and gather information about a situation people want to change to define a simple problem that can be solved through the development of a new or improved object or tool.
DETAILED DESCRIPTOR	K-2-ETS1-2.	Develop a simple sketch, drawing, or physical model to illustrate how the shape of an object helps it function as needed to solve a given problem.

Iowa Student Standards

Science

Grade **1** - Adopted: **2015**

STRAND / COURSE	IA.K-2-ETS1.	Engineering Design
ESSENTIAL CONCEPT AND/OR SKILL		Students who demonstrate understanding can:
DETAILED DESCRIPTOR	K-2-ETS1-1.	Ask questions, make observations, and gather information about a situation people want to change to define a simple problem that can be solved through the development of a new or improved object or tool.
DETAILED DESCRIPTOR	K-2-ETS1-2.	Develop a simple sketch, drawing, or physical model to illustrate how the shape of an object helps it function as needed to solve a given problem.

Iowa Student Standards

Science

Grade **2** - Adopted: **2015**

STRAND / COURSE	IA.2-PS1.	Matter and its Interactions
ESSENTIAL CONCEPT AND/OR SKILL		Students who demonstrate understanding can:
DETAILED DESCRIPTOR	2-PS1-2.	Analyze data obtained from testing different materials to determine which materials have the properties that are best suited for an intended purpose.
DETAILED DESCRIPTOR	2-PS1-3.	Make observations to construct an evidence-based account of how an object made of a small set of pieces can be disassembled and made into a new object.
STRAND / COURSE	IA.K-2-ETS1.	Engineering Design
ESSENTIAL CONCEPT AND/OR SKILL		Students who demonstrate understanding can:
DETAILED DESCRIPTOR	K-2-ETS1-1.	Ask questions, make observations, and gather information about a situation people want to change to define a simple problem that can be solved through the development of a new or improved object or tool.
DETAILED DESCRIPTOR	K-2-ETS1-2.	Develop a simple sketch, drawing, or physical model to illustrate how the shape of an object helps it function as needed to solve a given problem.

Iowa Student Standards

Science

Grade **3** - Adopted: **2015**

STRAND / COURSE	IA.3-PS2.	Motion and Stability: Forces and Interactions
ESSENTIAL CONCEPT AND/OR SKILL		Students who demonstrate understanding can:
DETAILED DESCRIPTOR	3-PS2-2.	Make observations and/or measurements of an object's motion to provide evidence that a pattern can be used to predict future motion.
STRAND / COURSE	IA.3-5-ETS1.	Engineering Design
ESSENTIAL CONCEPT AND/OR SKILL		Students who demonstrate understanding can:

DETAILED DESCRIPTOR	3-5-ETS1-1.	Define a simple design problem reflecting a need or a want that includes specified criteria for success and constraints on materials, time, or cost.
DETAILED DESCRIPTOR	3-5-ETS1-2.	Generate and compare multiple possible solutions to a problem based on how well each is likely to meet the criteria and constraints of the problem.
DETAILED DESCRIPTOR	3-5-ETS1-3.	Plan and carry out fair tests in which variables are controlled and failure points are considered to identify aspects of a model or prototype that can be improved.

Iowa Student Standards

Science

Grade **4** - Adopted: **2015**

STRAND / COURSE	IA.3-5-ETS1.	Engineering Design
ESSENTIAL CONCEPT AND/OR SKILL		Students who demonstrate understanding can:
DETAILED DESCRIPTOR	3-5-ETS1-1.	Define a simple design problem reflecting a need or a want that includes specified criteria for success and constraints on materials, time, or cost.
DETAILED DESCRIPTOR	3-5-ETS1-2.	Generate and compare multiple possible solutions to a problem based on how well each is likely to meet the criteria and constraints of the problem.
DETAILED DESCRIPTOR	3-5-ETS1-3.	Plan and carry out fair tests in which variables are controlled and failure points are considered to identify aspects of a model or prototype that can be improved.

Iowa Student Standards

Science

Grade **5** - Adopted: **2015**

STRAND / COURSE	IA.3-5-ETS1.	Engineering Design
ESSENTIAL CONCEPT AND/OR SKILL		Students who demonstrate understanding can:
DETAILED DESCRIPTOR	3-5-ETS1-1.	Define a simple design problem reflecting a need or a want that includes specified criteria for success and constraints on materials, time, or cost.
DETAILED DESCRIPTOR	3-5-ETS1-2.	Generate and compare multiple possible solutions to a problem based on how well each is likely to meet the criteria and constraints of the problem.
DETAILED DESCRIPTOR	3-5-ETS1-3.	Plan and carry out fair tests in which variables are controlled and failure points are considered to identify aspects of a model or prototype that can be improved.

Iowa Student Standards

Technology Education

Grade **K** - Adopted: **2018**

STRAND / COURSE		CSTA K-12 Computer Science Standards
ESSENTIAL CONCEPT AND/OR SKILL	CSTA.1A.	Level 1A (Ages 5-7)
DETAILED DESCRIPTOR	1A-CS.	Computing Systems
GRADE LEVEL EXPECTATION		Devices
EXAMPLE	1A-CS-01.	Select and operate appropriate software to perform a variety of tasks, and recognize that users have different needs and preferences for the technology they use. (P1.1)
STRAND / COURSE		CSTA K-12 Computer Science Standards
ESSENTIAL CONCEPT AND/OR SKILL	CSTA.1A.	Level 1A (Ages 5-7)
DETAILED DESCRIPTOR	1A-AP.	Algorithms & Programming
GRADE LEVEL EXPECTATION		Control
EXAMPLE	1A-AP-10.	Develop programs with sequences and simple loops, to express ideas or address a problem. (P5.2)

STRAND / COURSE		CSTA K-12 Computer Science Standards
ESSENTIAL CONCEPT AND/OR SKILL	CSTA.1A.	Level 1A (Ages 5-7)
DETAILED DESCRIPTOR	1A-AP.	Algorithms & Programming
GRADE LEVEL EXPECTATION		Program Development
EXAMPLE	1A-AP-12.	Develop plans that describe a program's sequence of events, goals, and expected outcomes. (P5.1, P7.2)
EXAMPLE	1A-AP-15.	Using correct terminology, describe steps taken and choices made during the iterative process of program development. (P7.2)

Iowa Student Standards

Technology Education

Grade **1** - Adopted: **2018**

STRAND / COURSE		CSTA K-12 Computer Science Standards
ESSENTIAL CONCEPT AND/OR SKILL	CSTA.1A.	Level 1A (Ages 5-7)
DETAILED DESCRIPTOR	1A-CS.	Computing Systems
GRADE LEVEL EXPECTATION		Devices
EXAMPLE	1A-CS-01.	Select and operate appropriate software to perform a variety of tasks, and recognize that users have different needs and preferences for the technology they use. (P1.1)
STRAND / COURSE		CSTA K-12 Computer Science Standards
ESSENTIAL CONCEPT AND/OR SKILL	CSTA.1A.	Level 1A (Ages 5-7)
DETAILED DESCRIPTOR	1A-AP.	Algorithms & Programming
GRADE LEVEL EXPECTATION		Control
EXAMPLE	1A-AP-10.	Develop programs with sequences and simple loops, to express ideas or address a problem. (P5.2)
STRAND / COURSE		CSTA K-12 Computer Science Standards
ESSENTIAL CONCEPT AND/OR SKILL	CSTA.1A.	Level 1A (Ages 5-7)
DETAILED DESCRIPTOR	1A-AP.	Algorithms & Programming
GRADE LEVEL EXPECTATION		Program Development
EXAMPLE	1A-AP-12.	Develop plans that describe a program's sequence of events, goals, and expected outcomes. (P5.1, P7.2)
EXAMPLE	1A-AP-15.	Using correct terminology, describe steps taken and choices made during the iterative process of program development. (P7.2)

Iowa Student Standards

Technology Education

Grade **2** - Adopted: **2018**

STRAND / COURSE		CSTA K-12 Computer Science Standards
ESSENTIAL CONCEPT AND/OR SKILL	CSTA.1A.	Level 1A (Ages 5-7)
DETAILED DESCRIPTOR	1A-CS.	Computing Systems

GRADE LEVEL EXPECTATION		Devices
EXAMPLE	1A-CS-01.	Select and operate appropriate software to perform a variety of tasks, and recognize that users have different needs and preferences for the technology they use. (P1.1)
STRAND / COURSE		CSTA K-12 Computer Science Standards
ESSENTIAL CONCEPT AND/OR SKILL	CSTA.1A.	Level 1A (Ages 5-7)
DETAILED DESCRIPTOR	1A-AP.	Algorithms & Programming
GRADE LEVEL EXPECTATION		Control
EXAMPLE	1A-AP-10.	Develop programs with sequences and simple loops, to express ideas or address a problem. (P5.2)
STRAND / COURSE		CSTA K-12 Computer Science Standards
ESSENTIAL CONCEPT AND/OR SKILL	CSTA.1A.	Level 1A (Ages 5-7)
DETAILED DESCRIPTOR	1A-AP.	Algorithms & Programming
GRADE LEVEL EXPECTATION		Program Development
EXAMPLE	1A-AP-12.	Develop plans that describe a program's sequence of events, goals, and expected outcomes. (P5.1, P7.2)
EXAMPLE	1A-AP-15.	Using correct terminology, describe steps taken and choices made during the iterative process of program development. (P7.2)

Iowa Student Standards

Technology Education

Grade **3** - Adopted: **2018**

STRAND / COURSE		CSTA K-12 Computer Science Standards
ESSENTIAL CONCEPT AND/OR SKILL	CSTA.1B.	Level 1B (Ages 8-11)
DETAILED DESCRIPTOR	1B-AP.	Algorithms & Programming
GRADE LEVEL EXPECTATION		Variables
EXAMPLE	1B-AP-09.	Create programs that use variables to store and modify data. (P5.2)
STRAND / COURSE		CSTA K-12 Computer Science Standards
ESSENTIAL CONCEPT AND/OR SKILL	CSTA.1B.	Level 1B (Ages 8-11)
DETAILED DESCRIPTOR	1B-AP.	Algorithms & Programming
GRADE LEVEL EXPECTATION		Control
EXAMPLE	1B-AP-10.	Create programs that include sequences, events, loops, and conditionals. (P5.2)
STRAND / COURSE		CSTA K-12 Computer Science Standards
ESSENTIAL CONCEPT AND/OR SKILL	CSTA.1B.	Level 1B (Ages 8-11)
DETAILED DESCRIPTOR	1B-AP.	Algorithms & Programming
GRADE LEVEL EXPECTATION		Program Development
EXAMPLE	1B-AP-	Use an iterative process to plan the development of a program by including others''

	13.	perspectives and considering user preferences. (P1.1, P5.1)

Iowa Student Standards

Technology Education

Grade **4** - Adopted: **2018**

STRAND / COURSE		CSTA K-12 Computer Science Standards
ESSENTIAL CONCEPT AND/OR SKILL	CSTA.1B.	Level 1B (Ages 8-11)
DETAILED DESCRIPTOR	1B-AP.	Algorithms & Programming
GRADE LEVEL EXPECTATION		Variables
EXAMPLE	1B-AP-09.	Create programs that use variables to store and modify data. (P5.2)
STRAND / COURSE		CSTA K-12 Computer Science Standards
ESSENTIAL CONCEPT AND/OR SKILL	CSTA.1B.	Level 1B (Ages 8-11)
DETAILED DESCRIPTOR	1B-AP.	Algorithms & Programming
GRADE LEVEL EXPECTATION		Control
EXAMPLE	1B-AP-10.	Create programs that include sequences, events, loops, and conditionals. (P5.2)
STRAND / COURSE		CSTA K-12 Computer Science Standards
ESSENTIAL CONCEPT AND/OR SKILL	CSTA.1B.	Level 1B (Ages 8-11)
DETAILED DESCRIPTOR	1B-AP.	Algorithms & Programming
GRADE LEVEL EXPECTATION		Program Development
EXAMPLE	1B-AP-13.	Use an iterative process to plan the development of a program by including others'' perspectives and considering user preferences. (P1.1, P5.1)

Iowa Student Standards

Technology Education

Grade **5** - Adopted: **2018**

STRAND / COURSE		CSTA K-12 Computer Science Standards
ESSENTIAL CONCEPT AND/OR SKILL	CSTA.1B.	Level 1B (Ages 8-11)
DETAILED DESCRIPTOR	1B-AP.	Algorithms & Programming
GRADE LEVEL EXPECTATION		Variables
EXAMPLE	1B-AP-09.	Create programs that use variables to store and modify data. (P5.2)
STRAND / COURSE		CSTA K-12 Computer Science Standards
ESSENTIAL CONCEPT AND/OR SKILL	CSTA.1B.	Level 1B (Ages 8-11)
DETAILED DESCRIPTOR	1B-AP.	Algorithms & Programming
GRADE LEVEL EXPECTATION		Control
EXAMPLE	1B-AP-	Create programs that include sequences, events, loops, and conditionals. (P5.2)

	10.	

STRAND / COURSE		CSTA K-12 Computer Science Standards
ESSENTIAL CONCEPT AND/OR SKILL	CSTA.1B.	Level 1B (Ages 8-11)
DETAILED DESCRIPTOR	1B-AP.	Algorithms & Programming
GRADE LEVEL EXPECTATION		Program Development
EXAMPLE	1B-AP-13.	Use an iterative process to plan the development of a program by including others'' perspectives and considering user preferences. (P1.1, P5.1)

Adventure with Lego: Lesson 5: Candy Town

Iowa Student Standards

Mathematics

Grade **K** - Adopted: **2012**

STRAND / COURSE	IA.CC.K.MD.	Measurement and Data
ESSENTIAL CONCEPT AND/OR SKILL		Describe and compare measurable attributes.
DETAILED DESCRIPTOR	K.MD.1.	Describe measurable attributes of objects, such as length or weight. Describe several measurable attributes of a single object. (K.MD.1.) (DOK 2)

Iowa Student Standards

Mathematics

Grade **2** - Adopted: **2012**

STRAND / COURSE	IA.CC.2.MD.	Measurement and Data
ESSENTIAL CONCEPT AND/OR SKILL		Measure and estimate lengths in standard units.
DETAILED DESCRIPTOR	2.MD.1.	Measure the length of an object by selecting and using appropriate tools such as rulers, yardsticks, meter sticks, and measuring tapes. (2.MD.1.) (DOK 1)
STRAND / COURSE	IA.CC.2.MD.	Measurement and Data
ESSENTIAL CONCEPT AND/OR SKILL		Represent and interpret data.
DETAILED DESCRIPTOR	2.MD.9.	Generate measurement data by measuring lengths of several objects to the nearest whole unit, or by making repeated measurements of the same object. Show the measurements by making a line plot, where the horizontal scale is marked off in whole-number units. (2.MD.9.) (DOK 2)

Iowa Student Standards

Mathematics

Grade **4** - Adopted: **2012**

STRAND / COURSE	IA.CC.4.MD.	Measurement and Data
ESSENTIAL CONCEPT AND/OR SKILL		Geometric measurement: understand concepts of angle and measure angles.
DETAILED DESCRIPTOR	4.MD.5.	Recognize angles as geometric shapes that are formed wherever two rays share a common endpoint, and understand concepts of angle measurement:
GRADE LEVEL EXPECTATION	4.MD.5.b.	An angle that turns through n one-degree angles is said to have an angle measure of n degrees. (4.MD.5.) (DOK 1)
STRAND / COURSE	IA.CC.4.MD.	Measurement and Data
ESSENTIAL CONCEPT AND/OR SKILL		Geometric measurement: understand concepts of angle and measure angles.

DETAILED DESCRIPTOR	4.MD.6.	Measure angles in whole-number degrees using a protractor. Sketch angles of specified measure. (4.MD.6.) (DOK 1)

Iowa Student Standards

Science

Grade **K** - Adopted: **2015**

STRAND / COURSE	IA.K-PS2.	Motion and Stability: Forces and interactions
ESSENTIAL CONCEPT AND/OR SKILL		Students who demonstrate understanding can:
DETAILED DESCRIPTOR	K-PS2-1.	Plan and conduct an investigation to compare the effects of different strengths or different directions of pushes and pulls on the motion of an object.
DETAILED DESCRIPTOR	K-PS2-2.	Analyze data to determine if a design solution works as intended to change the speed or direction of an object with a push or a pull.
STRAND / COURSE	IA.K-2-ETS1.	Engineering Design
ESSENTIAL CONCEPT AND/OR SKILL		Students who demonstrate understanding can:
DETAILED DESCRIPTOR	K-2-ETS1-1.	Ask questions, make observations, and gather information about a situation people want to change to define a simple problem that can be solved through the development of a new or improved object or tool.
DETAILED DESCRIPTOR	K-2-ETS1-2.	Develop a simple sketch, drawing, or physical model to illustrate how the shape of an object helps it function as needed to solve a given problem.

Iowa Student Standards

Science

Grade **1** - Adopted: **2015**

STRAND / COURSE	IA.K-2-ETS1.	Engineering Design
ESSENTIAL CONCEPT AND/OR SKILL		Students who demonstrate understanding can:
DETAILED DESCRIPTOR	K-2-ETS1-1.	Ask questions, make observations, and gather information about a situation people want to change to define a simple problem that can be solved through the development of a new or improved object or tool.
DETAILED DESCRIPTOR	K-2-ETS1-2.	Develop a simple sketch, drawing, or physical model to illustrate how the shape of an object helps it function as needed to solve a given problem.

Iowa Student Standards

Science

Grade **2** - Adopted: **2015**

STRAND / COURSE	IA.2-PS1.	Matter and its Interactions
ESSENTIAL CONCEPT AND/OR SKILL		Students who demonstrate understanding can:
DETAILED DESCRIPTOR	2-PS1-2.	Analyze data obtained from testing different materials to determine which materials have the properties that are best suited for an intended purpose.
DETAILED DESCRIPTOR	2-PS1-3.	Make observations to construct an evidence-based account of how an object made of a small set of pieces can be disassembled and made into a new object.
STRAND / COURSE	IA.K-2-ETS1.	Engineering Design
ESSENTIAL CONCEPT AND/OR SKILL		Students who demonstrate understanding can:
DETAILED DESCRIPTOR	K-2-ETS1-1.	Ask questions, make observations, and gather information about a situation people want to change to define a simple problem that can be solved through the development of a new or improved object or tool.
DETAILED DESCRIPTOR	K-2-ETS1-2.	Develop a simple sketch, drawing, or physical model to illustrate how the shape of an object helps it function as needed to solve a given problem.

Iowa Student Standards

Science

Grade **3** - Adopted: **2015**

STRAND / COURSE	IA.3-PS2.	Motion and Stability: Forces and Interactions
ESSENTIAL CONCEPT AND/OR SKILL		Students who demonstrate understanding can:
DETAILED DESCRIPTOR	3-PS2-1.	Plan and conduct an investigation to provide evidence of the effects of balanced and unbalanced forces on the motion of an object.
DETAILED DESCRIPTOR	3-PS2-2.	Make observations and/or measurements of an object's motion to provide evidence that a pattern can be used to predict future motion.
STRAND / COURSE	IA.3-5-ETS1.	Engineering Design
ESSENTIAL CONCEPT AND/OR SKILL		Students who demonstrate understanding can:
DETAILED DESCRIPTOR	3-5-ETS1-1.	Define a simple design problem reflecting a need or a want that includes specified criteria for success and constraints on materials, time, or cost.
DETAILED DESCRIPTOR	3-5-ETS1-2.	Generate and compare multiple possible solutions to a problem based on how well each is likely to meet the criteria and constraints of the problem.
DETAILED DESCRIPTOR	3-5-ETS1-3.	Plan and carry out fair tests in which variables are controlled and failure points are considered to identify aspects of a model or prototype that can be improved.

Iowa Student Standards

Science

Grade **4** - Adopted: **2015**

STRAND / COURSE	IA.4-PS3.	Energy
ESSENTIAL CONCEPT AND/OR SKILL		Students who demonstrate understanding can:
DETAILED DESCRIPTOR	4-PS3-1.	Use evidence to construct an explanation relating the speed of an object to the energy of that object.
DETAILED DESCRIPTOR	4-PS3-4.	Apply scientific ideas to design, test, and refine a device that converts energy from one form to another.
STRAND / COURSE	IA.3-5-ETS1.	Engineering Design
ESSENTIAL CONCEPT AND/OR SKILL		Students who demonstrate understanding can:
DETAILED DESCRIPTOR	3-5-ETS1-1.	Define a simple design problem reflecting a need or a want that includes specified criteria for success and constraints on materials, time, or cost.
DETAILED DESCRIPTOR	3-5-ETS1-2.	Generate and compare multiple possible solutions to a problem based on how well each is likely to meet the criteria and constraints of the problem.
DETAILED DESCRIPTOR	3-5-ETS1-3.	Plan and carry out fair tests in which variables are controlled and failure points are considered to identify aspects of a model or prototype that can be improved.

Iowa Student Standards

Science

Grade **5** - Adopted: **2015**

STRAND / COURSE	IA.3-5-ETS1.	Engineering Design
ESSENTIAL CONCEPT AND/OR SKILL		Students who demonstrate understanding can:
DETAILED DESCRIPTOR	3-5-ETS1-1.	Define a simple design problem reflecting a need or a want that includes specified criteria for success and constraints on materials, time, or cost.
DETAILED DESCRIPTOR	3-5-ETS1-2.	Generate and compare multiple possible solutions to a problem based on how well each is likely to meet the criteria and constraints of the problem.
DETAILED DESCRIPTOR	3-5-ETS1-3.	Plan and carry out fair tests in which variables are controlled and failure points are considered to identify aspects of a model or prototype that can be improved.

Iowa Student Standards

Technology Education

Grade **K** - Adopted: **2018**

STRAND / COURSE		CSTA K-12 Computer Science Standards
ESSENTIAL CONCEPT AND/OR SKILL	CSTA.1A.	Level 1A (Ages 5-7)
DETAILED DESCRIPTOR	1A-CS.	Computing Systems
GRADE LEVEL EXPECTATION		Devices
EXAMPLE	1A-CS-01.	Select and operate appropriate software to perform a variety of tasks, and recognize that users have different needs and preferences for the technology they use. (P1.1)
STRAND / COURSE		CSTA K-12 Computer Science Standards
ESSENTIAL CONCEPT AND/OR SKILL	CSTA.1A.	Level 1A (Ages 5-7)
DETAILED DESCRIPTOR	1A-AP.	Algorithms & Programming
GRADE LEVEL EXPECTATION		Control
EXAMPLE	1A-AP-10.	Develop programs with sequences and simple loops, to express ideas or address a problem. (P5.2)
STRAND / COURSE		CSTA K-12 Computer Science Standards
ESSENTIAL CONCEPT AND/OR SKILL	CSTA.1A.	Level 1A (Ages 5-7)
DETAILED DESCRIPTOR	1A-AP.	Algorithms & Programming
GRADE LEVEL EXPECTATION		Program Development
EXAMPLE	1A-AP-12.	Develop plans that describe a program’s sequence of events, goals, and expected outcomes. (P5.1, P7.2)
EXAMPLE	1A-AP-15.	Using correct terminology, describe steps taken and choices made during the iterative process of program development. (P7.2)

Iowa Student Standards

Technology Education

Grade **1** - Adopted: **2018**

STRAND / COURSE		CSTA K-12 Computer Science Standards
ESSENTIAL CONCEPT AND/OR SKILL	CSTA.1A.	Level 1A (Ages 5-7)
DETAILED DESCRIPTOR	1A-CS.	Computing Systems
GRADE LEVEL EXPECTATION		Devices
EXAMPLE	1A-CS-01.	Select and operate appropriate software to perform a variety of tasks, and recognize that users have different needs and preferences for the technology they use. (P1.1)
STRAND / COURSE		CSTA K-12 Computer Science Standards
ESSENTIAL CONCEPT AND/OR SKILL	CSTA.1A.	Level 1A (Ages 5-7)
DETAILED DESCRIPTOR	1A-AP.	Algorithms & Programming
GRADE LEVEL EXPECTATION		Control
EXAMPLE	1A-AP-	Develop programs with sequences and simple loops, to express ideas or address a

	10.	problem. (P5.2)

STRAND / COURSE		CSTA K-12 Computer Science Standards
ESSENTIAL CONCEPT AND/OR SKILL	CSTA.1A.	Level 1A (Ages 5-7)
DETAILED DESCRIPTOR	1A-AP.	Algorithms & Programming
GRADE LEVEL EXPECTATION		Program Development
EXAMPLE	1A-AP-12.	Develop plans that describe a program's sequence of events, goals, and expected outcomes. (P5.1, P7.2)
EXAMPLE	1A-AP-15.	Using correct terminology, describe steps taken and choices made during the iterative process of program development. (P7.2)

Iowa Student Standards

Technology Education

Grade **2** - Adopted: **2018**

STRAND / COURSE		CSTA K-12 Computer Science Standards
ESSENTIAL CONCEPT AND/OR SKILL	CSTA.1A.	Level 1A (Ages 5-7)
DETAILED DESCRIPTOR	1A-CS.	Computing Systems
GRADE LEVEL EXPECTATION		Devices
EXAMPLE	1A-CS-01.	Select and operate appropriate software to perform a variety of tasks, and recognize that users have different needs and preferences for the technology they use. (P1.1)

STRAND / COURSE		CSTA K-12 Computer Science Standards
ESSENTIAL CONCEPT AND/OR SKILL	CSTA.1A.	Level 1A (Ages 5-7)
DETAILED DESCRIPTOR	1A-AP.	Algorithms & Programming
GRADE LEVEL EXPECTATION		Control
EXAMPLE	1A-AP-10.	Develop programs with sequences and simple loops, to express ideas or address a problem. (P5.2)

STRAND / COURSE		CSTA K-12 Computer Science Standards
ESSENTIAL CONCEPT AND/OR SKILL	CSTA.1A.	Level 1A (Ages 5-7)
DETAILED DESCRIPTOR	1A-AP.	Algorithms & Programming
GRADE LEVEL EXPECTATION		Program Development
EXAMPLE	1A-AP-12.	Develop plans that describe a program's sequence of events, goals, and expected outcomes. (P5.1, P7.2)
EXAMPLE	1A-AP-15.	Using correct terminology, describe steps taken and choices made during the iterative process of program development. (P7.2)

Iowa Student Standards

Technology Education

Grade **3** - Adopted: **2018**

STRAND / COURSE		CSTA K-12 Computer Science Standards
ESSENTIAL CONCEPT AND/OR SKILL	CSTA.1B.	Level 1B (Ages 8-11)
DETAILED	1B-AP.	Algorithms & Programming

DETAILED DESCRIPTOR		
GRADE LEVEL EXPECTATION		Variables
EXAMPLE	1B-AP-09.	Create programs that use variables to store and modify data. (P5.2)
STRAND / COURSE		CSTA K-12 Computer Science Standards
ESSENTIAL CONCEPT AND/OR SKILL	CSTA.1B.	Level 1B (Ages 8-11)
DETAILED DESCRIPTOR	1B-AP.	Algorithms & Programming
GRADE LEVEL EXPECTATION		Control
EXAMPLE	1B-AP-10.	Create programs that include sequences, events, loops, and conditionals. (P5.2)
STRAND / COURSE		CSTA K-12 Computer Science Standards
ESSENTIAL CONCEPT AND/OR SKILL	CSTA.1B.	Level 1B (Ages 8-11)
DETAILED DESCRIPTOR	1B-AP.	Algorithms & Programming
GRADE LEVEL EXPECTATION		Program Development
EXAMPLE	1B-AP-13.	Use an iterative process to plan the development of a program by including others'' perspectives and considering user preferences. (P1.1, P5.1)

Iowa Student Standards

Technology Education

Grade **4** - Adopted: **2018**

STRAND / COURSE		CSTA K-12 Computer Science Standards
ESSENTIAL CONCEPT AND/OR SKILL	CSTA.1B.	Level 1B (Ages 8-11)
DETAILED DESCRIPTOR	1B-AP.	Algorithms & Programming
GRADE LEVEL EXPECTATION		Variables
EXAMPLE	1B-AP-09.	Create programs that use variables to store and modify data. (P5.2)
STRAND / COURSE		CSTA K-12 Computer Science Standards
ESSENTIAL CONCEPT AND/OR SKILL	CSTA.1B.	Level 1B (Ages 8-11)
DETAILED DESCRIPTOR	1B-AP.	Algorithms & Programming
GRADE LEVEL EXPECTATION		Control
EXAMPLE	1B-AP-10.	Create programs that include sequences, events, loops, and conditionals. (P5.2)
STRAND / COURSE		CSTA K-12 Computer Science Standards
ESSENTIAL CONCEPT AND/OR SKILL	CSTA.1B.	Level 1B (Ages 8-11)
DETAILED DESCRIPTOR	1B-AP.	Algorithms & Programming
GRADE LEVEL EXPECTATION		Program Development
EXAMPLE	1B-AP-13.	Use an iterative process to plan the development of a program by including others'' perspectives and considering user preferences. (P1.1, P5.1)

Iowa Student Standards

Technology Education

Grade **5** - Adopted: **2018**

STRAND / COURSE		CSTA K-12 Computer Science Standards
ESSENTIAL CONCEPT AND/OR SKILL	CSTA.1B.	Level 1B (Ages 8-11)
DETAILED DESCRIPTOR	1B-AP.	Algorithms & Programming
GRADE LEVEL EXPECTATION		Variables
EXAMPLE	1B-AP-09.	Create programs that use variables to store and modify data. (P5.2)
STRAND / COURSE		CSTA K-12 Computer Science Standards
ESSENTIAL CONCEPT AND/OR SKILL	CSTA.1B.	Level 1B (Ages 8-11)
DETAILED DESCRIPTOR	1B-AP.	Algorithms & Programming
GRADE LEVEL EXPECTATION		Control
EXAMPLE	1B-AP-10.	Create programs that include sequences, events, loops, and conditionals. (P5.2)
STRAND / COURSE		CSTA K-12 Computer Science Standards
ESSENTIAL CONCEPT AND/OR SKILL	CSTA.1B.	Level 1B (Ages 8-11)
DETAILED DESCRIPTOR	1B-AP.	Algorithms & Programming
GRADE LEVEL EXPECTATION		Program Development
EXAMPLE	1B-AP-13.	Use an iterative process to plan the development of a program by including others'' perspectives and considering user preferences. (P1.1, P5.1)

Adventure with Lego: Lesson 6: Sketch It!

Iowa Student Standards

Mathematics

Grade **K** - Adopted: **2012**

STRAND / COURSE	IA.CC.K.MD.	Measurement and Data
ESSENTIAL CONCEPT AND/OR SKILL		Describe and compare measurable attributes.
DETAILED DESCRIPTOR	K.MD.1.	Describe measurable attributes of objects, such as length or weight. Describe several measurable attributes of a single object. (K.MD.1.) (DOK 2)

Iowa Student Standards

Mathematics

Grade **1** - Adopted: **2012**

STRAND / COURSE	IA.CC.1.OA.	Operations and Algebraic Thinking
ESSENTIAL CONCEPT AND/OR SKILL		Add and subtract within 20.
DETAILED DESCRIPTOR	1.OA.6.	Add and subtract within 20, demonstrating fluency for addition and subtraction within 10. Use strategies such as counting on; making ten (e.g., 8 + 6 = 8 + 2 + 4 = 10 + 4 = 14); decomposing a number leading to a ten (e.g., 13 – 4 = 13 – 3 – 1 = 10 – 1 = 9);

		using the relationship between addition and subtraction (e.g., knowing that 8 + 4 = 12, one knows 12 – 8 = 4); and creating equivalent but easier or known sums (e.g., adding 6 + 7 by creating the known equivalent 6 + 6 + 1 = 12 + 1 = 13). (1.OA.6.) (DOK 1,2)
STRAND / COURSE	IA.CC.1.NBT.	Number and Operations in Base Ten
ESSENTIAL CONCEPT AND/OR SKILL		Use place value understanding and properties of operations to add and subtract.
DETAILED DESCRIPTOR	1.NBT.4.	Add within 100, including adding a two-digit number and a one-digit number, and adding a two-digit number and a multiple of 10, using concrete models or drawings and strategies based on place value, properties of operations, and/or the relationship between addition and subtraction; relate the strategy to a written method and explain the reasoning used. Understand that in adding two-digit numbers, one adds tens and tens, ones and ones; and sometimes it is necessary to compose a ten. (1.NBT.4.) (DOK 1,2,3)
STRAND / COURSE	IA.CC.1.G.	Geometry
ESSENTIAL CONCEPT AND/OR SKILL		Reason with shapes and their attributes.
DETAILED DESCRIPTOR	1.G.1.	Distinguish between defining attributes (e.g., triangles are closed and three-sided) versus non-defining attributes (e.g., color, orientation, overall size); build and draw shapes to possess defining attributes. (1.G.1.) (DOK 2)

Iowa Student Standards

Mathematics

Grade **2** - Adopted: **2012**

STRAND / COURSE	IA.CC.2.NBT.	Number and Operations in Base Ten
ESSENTIAL CONCEPT AND/OR SKILL		Use place value understanding and properties of operations to add and subtract.
DETAILED DESCRIPTOR	2.NBT.5.	Fluently add and subtract within 100 using strategies based on place value, properties of operations, and/or the relationship between addition and subtraction. (2.NBT.5.) (DOK 1,2)
DETAILED DESCRIPTOR	2.NBT.6.	Add up to four two-digit numbers using strategies based on place value and properties of operations. (2.NBT.6.) (DOK 2)
DETAILED DESCRIPTOR	2.NBT.7.	Add and subtract within 1000, using concrete models or drawings and strategies based on place value, properties of operations, and/or the relationship between addition and subtraction; relate the strategy to a written method. Understand that in adding or subtracting three-digit numbers, one adds or subtracts hundreds and hundreds, tens and tens, ones and ones; and sometimes it is necessary to compose or decompose tens or hundreds. (2.NBT.7.) (DOK 2)
DETAILED DESCRIPTOR	2.NBT.9.	Explain why addition and subtraction strategies work, using place value and the properties of operations. (2.NBT.9.) (DOK 3)
STRAND / COURSE	IA.CC.2.MD.	Measurement and Data
ESSENTIAL CONCEPT AND/OR SKILL		Measure and estimate lengths in standard units.
DETAILED DESCRIPTOR	2.MD.1.	Measure the length of an object by selecting and using appropriate tools such as rulers, yardsticks, meter sticks, and measuring tapes. (2.MD.1.) (DOK 1)
STRAND / COURSE	IA.CC.2.MD.	Measurement and Data
ESSENTIAL CONCEPT AND/OR SKILL		Represent and interpret data.
DETAILED DESCRIPTOR	2.MD.9.	Generate measurement data by measuring lengths of several objects to the nearest whole unit, or by making repeated measurements of the same object. Show the measurements by making a line plot, where the horizontal scale is marked off in whole-number units. (2.MD.9.) (DOK 2)

Iowa Student Standards

Mathematics

Grade **3** - Adopted: **2012**

STRAND / COURSE	IA.CC.3.OA.	Operations and Algebraic Thinking
ESSENTIAL CONCEPT AND/OR SKILL		Multiply and divide within 100.
DETAILED DESCRIPTOR	3.OA.7.	Fluently multiply and divide within 100, using strategies such as the relationship between multiplication and division (e.g., knowing that 8 × 5 = 40, one knows 40 ÷ 5 = 8) or properties of operations. By the end of Grade 3, know from memory all products of two one-digit numbers. (3.OA.7.) (DOK 1,2)
STRAND / COURSE	IA.CC.3.NBT.	Number and Operations in Base Ten
ESSENTIAL CONCEPT AND/OR SKILL		Use place value understanding and properties of operations to perform multi-digit arithmetic.
DETAILED DESCRIPTOR	3.NBT.2.	Fluently add and subtract within 1000 using strategies and algorithms based on place value, properties of operations, and/or the relationship between addition and subtraction. (3.NBT.2.) (DOK 1,2)

Iowa Student Standards

Mathematics

Grade **4** - Adopted: **2012**

STRAND / COURSE	IA.CC.4.NBT.	Number and Operations in Base Ten
ESSENTIAL CONCEPT AND/OR SKILL		Use place value understanding and properties of operations to perform multi-digit arithmetic.
DETAILED DESCRIPTOR	4.NBT.4.	Fluently add and subtract multi-digit whole numbers using the standard algorithm. (4.NBT.4.) (DOK 1)
DETAILED DESCRIPTOR	4.NBT.6.	Find whole-number quotients and remainders with up to four-digit dividends and one-digit divisors, using strategies based on place value, the properties of operations, and/or the relationship between multiplication and division. Illustrate and explain the calculation by using equations, rectangular arrays, and/or area models. (4.NBT.6.) (DOK 1,2)
STRAND / COURSE	IA.CC.4.MD.	Measurement and Data
ESSENTIAL CONCEPT AND/OR SKILL		Geometric measurement: understand concepts of angle and measure angles.
DETAILED DESCRIPTOR	4.MD.5.	Recognize angles as geometric shapes that are formed wherever two rays share a common endpoint, and understand concepts of angle measurement:
GRADE LEVEL EXPECTATION	4.MD.5.a.	An angle is measured with reference to a circle with its center at the common endpoint of the rays, by considering the fraction of the circular arc between the points where the two rays intersect the circle. An angle that turns through 1/360 of a circle is called a "one-degree angle," and can be used to measure angles. (4.MD.5.) (DOK 1)
GRADE LEVEL EXPECTATION	4.MD.5.b.	An angle that turns through n one-degree angles is said to have an angle measure of n degrees. (4.MD.5.) (DOK 1)
STRAND / COURSE	IA.CC.4.MD.	Measurement and Data
ESSENTIAL CONCEPT AND/OR SKILL		Geometric measurement: understand concepts of angle and measure angles.
DETAILED DESCRIPTOR	4.MD.6.	Measure angles in whole-number degrees using a protractor. Sketch angles of specified measure. (4.MD.6.) (DOK 1)
DETAILED DESCRIPTOR	4.MD.7.	Recognize angle measure as additive. When an angle is decomposed into non-overlapping parts, the angle measure of the whole is the sum of the angle measures of the parts. Solve addition and subtraction problems to find unknown angles on a diagram in real world and mathematical problems, e.g., by using an equation with a symbol for the unknown angle measure. (4.MD.7.) (DOK 1,2)
STRAND / COURSE	IA.CC.4.G.	Geometry
ESSENTIAL CONCEPT AND/OR SKILL		Draw and identify lines and angles, and classify shapes by properties of their lines and angles.
DETAILED DESCRIPTOR	4.G.1.	Draw points, lines, line segments, rays, angles (right, acute, obtuse), and perpendicular and parallel lines. Identify these in two-dimensional figures. (4.G.1.)

		(DOK 1)

Iowa Student Standards

Mathematics

Grade **5** - Adopted: **2012**

STRAND / COURSE	**IA.CC.5. NBT.**	**Number and Operations in Base Ten**
ESSENTIAL CONCEPT AND/OR SKILL		**Perform operations with multi-digit whole numbers and with decimals to hundredths.**
DETAILED DESCRIPTOR	**5.NBT.6.**	**Find whole-number quotients of whole numbers with up to four-digit dividends and two-digit divisors, using strategies based on place value, the properties of operations, and/or the relationship between multiplication and division. Illustrate and explain the calculation by using equations, rectangular arrays, and/or area models. (5.NBT.6.) (DOK 1,2)**

Iowa Student Standards

Science

Grade **K** - Adopted: **2015**

STRAND / COURSE	**IA.K-PS2.**	**Motion and Stability: Forces and interactions**
ESSENTIAL CONCEPT AND/OR SKILL		**Students who demonstrate understanding can:**
DETAILED DESCRIPTOR	**K-PS2-1.**	**Plan and conduct an investigation to compare the effects of different strengths or different directions of pushes and pulls on the motion of an object.**
STRAND / COURSE	**IA.K-2-ETS1.**	**Engineering Design**
ESSENTIAL CONCEPT AND/OR SKILL		**Students who demonstrate understanding can:**
DETAILED DESCRIPTOR	**K-2-ETS1-1.**	**Ask questions, make observations, and gather information about a situation people want to change to define a simple problem that can be solved through the development of a new or improved object or tool.**
DETAILED DESCRIPTOR	**K-2-ETS1-2.**	**Develop a simple sketch, drawing, or physical model to illustrate how the shape of an object helps it function as needed to solve a given problem.**

Iowa Student Standards

Science

Grade **1** - Adopted: **2015**

STRAND / COURSE	**IA.K-2-ETS1.**	**Engineering Design**
ESSENTIAL CONCEPT AND/OR SKILL		**Students who demonstrate understanding can:**
DETAILED DESCRIPTOR	**K-2-ETS1-1.**	**Ask questions, make observations, and gather information about a situation people want to change to define a simple problem that can be solved through the development of a new or improved object or tool.**
DETAILED DESCRIPTOR	**K-2-ETS1-2.**	**Develop a simple sketch, drawing, or physical model to illustrate how the shape of an object helps it function as needed to solve a given problem.**

Iowa Student Standards

Science

Grade **2** - Adopted: **2015**

STRAND / COURSE	**IA.2-PS1.**	**Matter and its Interactions**
ESSENTIAL CONCEPT AND/OR SKILL		**Students who demonstrate understanding can:**
DETAILED DESCRIPTOR	**2-PS1-2.**	**Analyze data obtained from testing different materials to determine which materials have the properties that are best suited for an intended purpose.**
DETAILED DESCRIPTOR	**2-PS1-3.**	**Make observations to construct an evidence-based account of how an object made of a small set of pieces can be disassembled and made into a new object.**

STRAND / COURSE	IA.K-2-ETS1.	Engineering Design
ESSENTIAL CONCEPT AND/OR SKILL		Students who demonstrate understanding can:
DETAILED DESCRIPTOR	K-2-ETS1-1.	Ask questions, make observations, and gather information about a situation people want to change to define a simple problem that can be solved through the development of a new or improved object or tool.
DETAILED DESCRIPTOR	K-2-ETS1-2.	Develop a simple sketch, drawing, or physical model to illustrate how the shape of an object helps it function as needed to solve a given problem.

Iowa Student Standards

Science

Grade **3** - Adopted: **2015**

STRAND / COURSE	IA.3-PS2.	Motion and Stability: Forces and Interactions
ESSENTIAL CONCEPT AND/OR SKILL		Students who demonstrate understanding can:
DETAILED DESCRIPTOR	3-PS2-2.	Make observations and/or measurements of an object's motion to provide evidence that a pattern can be used to predict future motion.
STRAND / COURSE	IA.3-5-ETS1.	Engineering Design
ESSENTIAL CONCEPT AND/OR SKILL		Students who demonstrate understanding can:
DETAILED DESCRIPTOR	3-5-ETS1-1.	Define a simple design problem reflecting a need or a want that includes specified criteria for success and constraints on materials, time, or cost.
DETAILED DESCRIPTOR	3-5-ETS1-2.	Generate and compare multiple possible solutions to a problem based on how well each is likely to meet the criteria and constraints of the problem.
DETAILED DESCRIPTOR	3-5-ETS1-3.	Plan and carry out fair tests in which variables are controlled and failure points are considered to identify aspects of a model or prototype that can be improved.

Iowa Student Standards

Science

Grade **4** - Adopted: **2015**

STRAND / COURSE	IA.3-5-ETS1.	Engineering Design
ESSENTIAL CONCEPT AND/OR SKILL		Students who demonstrate understanding can:
DETAILED DESCRIPTOR	3-5-ETS1-1.	Define a simple design problem reflecting a need or a want that includes specified criteria for success and constraints on materials, time, or cost.
DETAILED DESCRIPTOR	3-5-ETS1-2.	Generate and compare multiple possible solutions to a problem based on how well each is likely to meet the criteria and constraints of the problem.
DETAILED DESCRIPTOR	3-5-ETS1-3.	Plan and carry out fair tests in which variables are controlled and failure points are considered to identify aspects of a model or prototype that can be improved.

Iowa Student Standards

Science

Grade **5** - Adopted: **2015**

STRAND / COURSE	IA.3-5-ETS1.	Engineering Design
ESSENTIAL CONCEPT AND/OR SKILL		Students who demonstrate understanding can:
DETAILED DESCRIPTOR	3-5-ETS1-1.	Define a simple design problem reflecting a need or a want that includes specified criteria for success and constraints on materials, time, or cost.
DETAILED DESCRIPTOR	3-5-ETS1-2.	Generate and compare multiple possible solutions to a problem based on how well each is likely to meet the criteria and constraints of the problem.
DETAILED DESCRIPTOR	3-5-ETS1-3.	Plan and carry out fair tests in which variables are controlled and failure points are considered to identify aspects of a model or prototype that can be improved.

Iowa Student Standards

Technology Education

Grade **K** - Adopted: **2018**

STRAND / COURSE		CSTA K-12 Computer Science Standards
ESSENTIAL CONCEPT AND/OR SKILL	CSTA.1A.	Level 1A (Ages 5-7)
DETAILED DESCRIPTOR	1A-CS.	Computing Systems
GRADE LEVEL EXPECTATION		Devices
EXAMPLE	1A-CS-01.	Select and operate appropriate software to perform a variety of tasks, and recognize that users have different needs and preferences for the technology they use. (P1.1)
STRAND / COURSE		CSTA K-12 Computer Science Standards
ESSENTIAL CONCEPT AND/OR SKILL	CSTA.1A.	Level 1A (Ages 5-7)
DETAILED DESCRIPTOR	1A-AP.	Algorithms & Programming
GRADE LEVEL EXPECTATION		Control
EXAMPLE	1A-AP-10.	Develop programs with sequences and simple loops, to express ideas or address a problem. (P5.2)
STRAND / COURSE		CSTA K-12 Computer Science Standards
ESSENTIAL CONCEPT AND/OR SKILL	CSTA.1A.	Level 1A (Ages 5-7)
DETAILED DESCRIPTOR	1A-AP.	Algorithms & Programming
GRADE LEVEL EXPECTATION		Program Development
EXAMPLE	1A-AP-12.	Develop plans that describe a program's sequence of events, goals, and expected outcomes. (P5.1, P7.2)
EXAMPLE	1A-AP-14.	Debug (identify and fix) errors in an algorithm or program that includes sequences and simple loops. (P6.2)
EXAMPLE	1A-AP-15.	Using correct terminology, describe steps taken and choices made during the iterative process of program development. (P7.2)

Iowa Student Standards

Technology Education

Grade **1** - Adopted: **2018**

STRAND / COURSE		CSTA K-12 Computer Science Standards
ESSENTIAL CONCEPT AND/OR SKILL	CSTA.1A.	Level 1A (Ages 5-7)
DETAILED DESCRIPTOR	1A-CS.	Computing Systems
GRADE LEVEL EXPECTATION		Devices
EXAMPLE	1A-CS-01.	Select and operate appropriate software to perform a variety of tasks, and recognize that users have different needs and preferences for the technology they use. (P1.1)
STRAND / COURSE		CSTA K-12 Computer Science Standards
ESSENTIAL CONCEPT AND/OR SKILL	CSTA.1A.	Level 1A (Ages 5-7)
DETAILED DESCRIPTOR	1A-AP.	Algorithms & Programming
GRADE LEVEL		Control

EXPECTATION		
EXAMPLE	1A-AP-10.	Develop programs with sequences and simple loops, to express ideas or address a problem. (P5.2)
STRAND / COURSE		CSTA K-12 Computer Science Standards
ESSENTIAL CONCEPT AND/OR SKILL	CSTA.1A.	Level 1A (Ages 5-7)
DETAILED DESCRIPTOR	1A-AP.	Algorithms & Programming
GRADE LEVEL EXPECTATION		Program Development
EXAMPLE	1A-AP-12.	Develop plans that describe a program's sequence of events, goals, and expected outcomes. (P5.1, P7.2)
EXAMPLE	1A-AP-14.	Debug (identify and fix) errors in an algorithm or program that includes sequences and simple loops. (P6.2)
EXAMPLE	1A-AP-15.	Using correct terminology, describe steps taken and choices made during the iterative process of program development. (P7.2)

Iowa Student Standards

Technology Education

Grade **2** - Adopted: **2018**

STRAND / COURSE		CSTA K-12 Computer Science Standards
ESSENTIAL CONCEPT AND/OR SKILL	CSTA.1A.	Level 1A (Ages 5-7)
DETAILED DESCRIPTOR	1A-CS.	Computing Systems
GRADE LEVEL EXPECTATION		Devices
EXAMPLE	1A-CS-01.	Select and operate appropriate software to perform a variety of tasks, and recognize that users have different needs and preferences for the technology they use. (P1.1)
STRAND / COURSE		CSTA K-12 Computer Science Standards
ESSENTIAL CONCEPT AND/OR SKILL	CSTA.1A.	Level 1A (Ages 5-7)
DETAILED DESCRIPTOR	1A-AP.	Algorithms & Programming
GRADE LEVEL EXPECTATION		Control
EXAMPLE	1A-AP-10.	Develop programs with sequences and simple loops, to express ideas or address a problem. (P5.2)
STRAND / COURSE		CSTA K-12 Computer Science Standards
ESSENTIAL CONCEPT AND/OR SKILL	CSTA.1A.	Level 1A (Ages 5-7)
DETAILED DESCRIPTOR	1A-AP.	Algorithms & Programming
GRADE LEVEL EXPECTATION		Program Development
EXAMPLE	1A-AP-12.	Develop plans that describe a program's sequence of events, goals, and expected outcomes. (P5.1, P7.2)
EXAMPLE	1A-AP-14.	Debug (identify and fix) errors in an algorithm or program that includes sequences and simple loops. (P6.2)
EXAMPLE	1A-AP-15.	Using correct terminology, describe steps taken and choices made during the iterative process of program development. (P7.2)

Iowa Student Standards

Technology Education

Grade **3** - Adopted: **2018**

STRAND / COURSE		CSTA K-12 Computer Science Standards
ESSENTIAL CONCEPT AND/OR SKILL	CSTA.1B.	Level 1B (Ages 8-11)
DETAILED DESCRIPTOR	1B-AP.	Algorithms & Programming
GRADE LEVEL EXPECTATION		Variables
EXAMPLE	1B-AP-09.	Create programs that use variables to store and modify data. (P5.2)

STRAND / COURSE		CSTA K-12 Computer Science Standards
ESSENTIAL CONCEPT AND/OR SKILL	CSTA.1B.	Level 1B (Ages 8-11)
DETAILED DESCRIPTOR	1B-AP.	Algorithms & Programming
GRADE LEVEL EXPECTATION		Control
EXAMPLE	1B-AP-10.	Create programs that include sequences, events, loops, and conditionals. (P5.2)

STRAND / COURSE		CSTA K-12 Computer Science Standards
ESSENTIAL CONCEPT AND/OR SKILL	CSTA.1B.	Level 1B (Ages 8-11)
DETAILED DESCRIPTOR	1B-AP.	Algorithms & Programming
GRADE LEVEL EXPECTATION		Program Development
EXAMPLE	1B-AP-13.	Use an iterative process to plan the development of a program by including others'' perspectives and considering user preferences. (P1.1, P5.1)
EXAMPLE	1B-AP-15.	Test and debug (identify and fix errors) a program or algorithm to ensure it runs as intended. (P6.1, P6.2)

Iowa Student Standards

Technology Education

Grade **4** - Adopted: **2018**

STRAND / COURSE		CSTA K-12 Computer Science Standards
ESSENTIAL CONCEPT AND/OR SKILL	CSTA.1B.	Level 1B (Ages 8-11)
DETAILED DESCRIPTOR	1B-AP.	Algorithms & Programming
GRADE LEVEL EXPECTATION		Variables
EXAMPLE	1B-AP-09.	Create programs that use variables to store and modify data. (P5.2)

STRAND / COURSE		CSTA K-12 Computer Science Standards
ESSENTIAL CONCEPT AND/OR SKILL	CSTA.1B.	Level 1B (Ages 8-11)
DETAILED DESCRIPTOR	1B-AP.	Algorithms & Programming
GRADE LEVEL EXPECTATION		Control
EXAMPLE	1B-AP-10.	Create programs that include sequences, events, loops, and conditionals. (P5.2)

STRAND / COURSE		CSTA K-12 Computer Science Standards

ESSENTIAL CONCEPT AND/OR SKILL	CSTA.1B.	Level 1B (Ages 8-11)
DETAILED DESCRIPTOR	1B-AP.	Algorithms & Programming
GRADE LEVEL EXPECTATION		Program Development
EXAMPLE	1B-AP-13.	Use an iterative process to plan the development of a program by including others'' perspectives and considering user preferences. (P1.1, P5.1)
EXAMPLE	1B-AP-15.	Test and debug (identify and fix errors) a program or algorithm to ensure it runs as intended. (P6.1, P6.2)

Iowa Student Standards

Technology Education

Grade **5** - Adopted: **2018**

STRAND / COURSE		CSTA K-12 Computer Science Standards
ESSENTIAL CONCEPT AND/OR SKILL	CSTA.1B.	Level 1B (Ages 8-11)
DETAILED DESCRIPTOR	1B-AP.	Algorithms & Programming
GRADE LEVEL EXPECTATION		Variables
EXAMPLE	1B-AP-09.	Create programs that use variables to store and modify data. (P5.2)
STRAND / COURSE		CSTA K-12 Computer Science Standards
ESSENTIAL CONCEPT AND/OR SKILL	CSTA.1B.	Level 1B (Ages 8-11)
DETAILED DESCRIPTOR	1B-AP.	Algorithms & Programming
GRADE LEVEL EXPECTATION		Control
EXAMPLE	1B-AP-10.	Create programs that include sequences, events, loops, and conditionals. (P5.2)
STRAND / COURSE		CSTA K-12 Computer Science Standards
ESSENTIAL CONCEPT AND/OR SKILL	CSTA.1B.	Level 1B (Ages 8-11)
DETAILED DESCRIPTOR	1B-AP.	Algorithms & Programming
GRADE LEVEL EXPECTATION		Program Development
EXAMPLE	1B-AP-13.	Use an iterative process to plan the development of a program by including others'' perspectives and considering user preferences. (P1.1, P5.1)
EXAMPLE	1B-AP-15.	Test and debug (identify and fix errors) a program or algorithm to ensure it runs as intended. (P6.1, P6.2)

Adventure with Lego: Lesson 7: The Milky Way

Iowa Student Standards

Mathematics

Grade **K** - Adopted: **2012**

STRAND / COURSE	IA.CC.K.MD.	Measurement and Data
ESSENTIAL CONCEPT AND/OR SKILL		Describe and compare measurable attributes.
DETAILED DESCRIPTOR	K.MD.1.	Describe measurable attributes of objects, such as length or weight. Describe several measurable attributes of a single object. (K.MD.1.) (DOK 2)

Iowa Student Standards

Mathematics

Grade **1** - Adopted: **2012**

STRAND / COURSE	IA.CC.1.OA.	Operations and Algebraic Thinking
ESSENTIAL CONCEPT AND/OR SKILL		Add and subtract within 20.
DETAILED DESCRIPTOR	1.OA.6.	Add and subtract within 20, demonstrating fluency for addition and subtraction within 10. Use strategies such as counting on; making ten (e.g., 8 + 6 = 8 + 2 + 4 = 10 + 4 = 14); decomposing a number leading to a ten (e.g., 13 – 4 = 13 – 3 – 1 = 10 – 1 = 9); using the relationship between addition and subtraction (e.g., knowing that 8 + 4 = 12, one knows 12 – 8 = 4); and creating equivalent but easier or known sums (e.g., adding 6 + 7 by creating the known equivalent 6 + 6 + 1 = 12 + 1 = 13). (1.OA.6.) (DOK 1,2)
STRAND / COURSE	IA.CC.1.NBT.	Number and Operations in Base Ten
ESSENTIAL CONCEPT AND/OR SKILL		Use place value understanding and properties of operations to add and subtract.
DETAILED DESCRIPTOR	1.NBT.4.	Add within 100, including adding a two-digit number and a one-digit number, and adding a two-digit number and a multiple of 10, using concrete models or drawings and strategies based on place value, properties of operations, and/or the relationship between addition and subtraction; relate the strategy to a written method and explain the reasoning used. Understand that in adding two-digit numbers, one adds tens and tens, ones and ones; and sometimes it is necessary to compose a ten. (1.NBT.4.) (DOK 1,2,3)
STRAND / COURSE	IA.CC.1.G.	Geometry
ESSENTIAL CONCEPT AND/OR SKILL		Reason with shapes and their attributes.
DETAILED DESCRIPTOR	1.G.1.	Distinguish between defining attributes (e.g., triangles are closed and three-sided) versus non-defining attributes (e.g., color, orientation, overall size); build and draw shapes to possess defining attributes. (1.G.1.) (DOK 2)

Iowa Student Standards

Mathematics

Grade **2** - Adopted: **2012**

STRAND / COURSE	IA.CC.2.NBT.	Number and Operations in Base Ten
ESSENTIAL CONCEPT AND/OR SKILL		Use place value understanding and properties of operations to add and subtract.
DETAILED DESCRIPTOR	2.NBT.5.	Fluently add and subtract within 100 using strategies based on place value, properties of operations, and/or the relationship between addition and subtraction. (2.NBT.5.) (DOK 1,2)
DETAILED DESCRIPTOR	2.NBT.6.	Add up to four two-digit numbers using strategies based on place value and properties of operations. (2.NBT.6.) (DOK 2)
DETAILED DESCRIPTOR	2.NBT.7.	Add and subtract within 1000, using concrete models or drawings and strategies based on place value, properties of operations, and/or the relationship between addition and subtraction; relate the strategy to a written method. Understand that in adding or subtracting three-digit numbers, one adds or subtracts hundreds and hundreds, tens and tens, ones and ones; and sometimes it is necessary to compose or decompose tens or hundreds. (2.NBT.7.) (DOK 2)
DETAILED DESCRIPTOR	2.NBT.9.	Explain why addition and subtraction strategies work, using place value and the properties of operations. (2.NBT.9.) (DOK 3)
STRAND / COURSE	IA.CC.2.MD.	Measurement and Data
ESSENTIAL CONCEPT AND/OR SKILL		Measure and estimate lengths in standard units.
DETAILED DESCRIPTOR	2.MD.1.	Measure the length of an object by selecting and using appropriate tools such as rulers, yardsticks, meter sticks, and measuring tapes. (2.MD.1.) (DOK 1)
STRAND /	IA.CC.2.	Measurement and Data

COURSE	MD.	
ESSENTIAL CONCEPT AND/OR SKILL		Represent and interpret data.
DETAILED DESCRIPTOR	2.MD.9.	Generate measurement data by measuring lengths of several objects to the nearest whole unit, or by making repeated measurements of the same object. Show the measurements by making a line plot, where the horizontal scale is marked off in whole-number units. (2.MD.9.) (DOK 2)

Iowa Student Standards

Mathematics

Grade **3** - Adopted: **2012**

STRAND / COURSE	IA.CC.3. OA.	Operations and Algebraic Thinking
ESSENTIAL CONCEPT AND/OR SKILL		Multiply and divide within 100.
DETAILED DESCRIPTOR	3.OA.7.	Fluently multiply and divide within 100, using strategies such as the relationship between multiplication and division (e.g., knowing that 8 × 5 = 40, one knows 40 ÷ 5 = 8) or properties of operations. By the end of Grade 3, know from memory all products of two one-digit numbers. (3.OA.7.) (DOK 1,2)
STRAND / COURSE	IA.CC.3. NBT.	Number and Operations in Base Ten
ESSENTIAL CONCEPT AND/OR SKILL		Use place value understanding and properties of operations to perform multi-digit arithmetic.
DETAILED DESCRIPTOR	3.NBT.2.	Fluently add and subtract within 1000 using strategies and algorithms based on place value, properties of operations, and/or the relationship between addition and subtraction. (3.NBT.2.) (DOK 1,2)

Iowa Student Standards

Mathematics

Grade **4** - Adopted: **2012**

STRAND / COURSE	IA.CC.4. NBT.	Number and Operations in Base Ten
ESSENTIAL CONCEPT AND/OR SKILL		Use place value understanding and properties of operations to perform multi-digit arithmetic.
DETAILED DESCRIPTOR	4.NBT.4.	Fluently add and subtract multi-digit whole numbers using the standard algorithm. (4.NBT.4.) (DOK 1)
DETAILED DESCRIPTOR	4.NBT.6.	Find whole-number quotients and remainders with up to four-digit dividends and one-digit divisors, using strategies based on place value, the properties of operations, and/or the relationship between multiplication and division. Illustrate and explain the calculation by using equations, rectangular arrays, and/or area models. (4.NBT.6.) (DOK 1,2)
STRAND / COURSE	IA.CC.4. MD.	Measurement and Data
ESSENTIAL CONCEPT AND/OR SKILL		Geometric measurement: understand concepts of angle and measure angles.
DETAILED DESCRIPTOR	4.MD.5.	Recognize angles as geometric shapes that are formed wherever two rays share a common endpoint, and understand concepts of angle measurement:
GRADE LEVEL EXPECTATION	4.MD.5.a.	An angle is measured with reference to a circle with its center at the common endpoint of the rays, by considering the fraction of the circular arc between the points where the two rays intersect the circle. An angle that turns through 1/360 of a circle is called a "one-degree angle," and can be used to measure angles. (4.MD.5.) (DOK 1)
GRADE LEVEL EXPECTATION	4.MD.5.b.	An angle that turns through n one-degree angles is said to have an angle measure of n degrees. (4.MD.5.) (DOK 1)
STRAND / COURSE	IA.CC.4. MD.	Measurement and Data
ESSENTIAL CONCEPT AND/OR SKILL		Geometric measurement: understand concepts of angle and measure angles.
DETAILED	4.MD.6.	Measure angles in whole-number degrees using a protractor. Sketch angles of

DESCRIPTOR		specified measure. (4.MD.6.) (DOK 1)
DETAILED DESCRIPTOR	4.MD.7.	Recognize angle measure as additive. When an angle is decomposed into non-overlapping parts, the angle measure of the whole is the sum of the angle measures of the parts. Solve addition and subtraction problems to find unknown angles on a diagram in real world and mathematical problems, e.g., by using an equation with a symbol for the unknown angle measure. (4.MD.7.) (DOK 1,2)
STRAND / COURSE	IA.CC.4.G.	Geometry
ESSENTIAL CONCEPT AND/OR SKILL		Draw and identify lines and angles, and classify shapes by properties of their lines and angles.
DETAILED DESCRIPTOR	4.G.1.	Draw points, lines, line segments, rays, angles (right, acute, obtuse), and perpendicular and parallel lines. Identify these in two-dimensional figures. (4.G.1.) (DOK 1)

Iowa Student Standards

Mathematics

Grade **5** - Adopted: **2012**

STRAND / COURSE	IA.CC.5.NBT.	Number and Operations in Base Ten
ESSENTIAL CONCEPT AND/OR SKILL		Perform operations with multi-digit whole numbers and with decimals to hundredths.
DETAILED DESCRIPTOR	5.NBT.6.	Find whole-number quotients of whole numbers with up to four-digit dividends and two-digit divisors, using strategies based on place value, the properties of operations, and/or the relationship between multiplication and division. Illustrate and explain the calculation by using equations, rectangular arrays, and/or area models. (5.NBT.6.) (DOK 1,2)

Iowa Student Standards

Science

Grade **K** - Adopted: **2015**

STRAND / COURSE	IA.K-PS2.	Motion and Stability: Forces and interactions
ESSENTIAL CONCEPT AND/OR SKILL		Students who demonstrate understanding can:
DETAILED DESCRIPTOR	K-PS2-1.	Plan and conduct an investigation to compare the effects of different strengths or different directions of pushes and pulls on the motion of an object.
STRAND / COURSE	IA.K-2-ETS1.	Engineering Design
ESSENTIAL CONCEPT AND/OR SKILL		Students who demonstrate understanding can:
DETAILED DESCRIPTOR	K-2-ETS1-1.	Ask questions, make observations, and gather information about a situation people want to change to define a simple problem that can be solved through the development of a new or improved object or tool.
DETAILED DESCRIPTOR	K-2-ETS1-2.	Develop a simple sketch, drawing, or physical model to illustrate how the shape of an object helps it function as needed to solve a given problem.

Iowa Student Standards

Science

Grade **1** - Adopted: **2015**

STRAND / COURSE	IA.K-2-ETS1.	Engineering Design
ESSENTIAL CONCEPT AND/OR SKILL		Students who demonstrate understanding can:
DETAILED DESCRIPTOR	K-2-ETS1-1.	Ask questions, make observations, and gather information about a situation people want to change to define a simple problem that can be solved through the development of a new or improved object or tool.
DETAILED DESCRIPTOR	K-2-ETS1-2.	Develop a simple sketch, drawing, or physical model to illustrate how the shape of an object helps it function as needed to solve a given problem.

Iowa Student Standards

Science

Grade **2** - Adopted: **2015**

STRAND / COURSE	IA.2-PS1.	Matter and its Interactions
ESSENTIAL CONCEPT AND/OR SKILL		Students who demonstrate understanding can:
DETAILED DESCRIPTOR	2-PS1-2.	Analyze data obtained from testing different materials to determine which materials have the properties that are best suited for an intended purpose.
DETAILED DESCRIPTOR	2-PS1-3.	Make observations to construct an evidence-based account of how an object made of a small set of pieces can be disassembled and made into a new object.
STRAND / COURSE	IA.K-2-ETS1.	Engineering Design
ESSENTIAL CONCEPT AND/OR SKILL		Students who demonstrate understanding can:
DETAILED DESCRIPTOR	K-2-ETS1-1.	Ask questions, make observations, and gather information about a situation people want to change to define a simple problem that can be solved through the development of a new or improved object or tool.
DETAILED DESCRIPTOR	K-2-ETS1-2.	Develop a simple sketch, drawing, or physical model to illustrate how the shape of an object helps it function as needed to solve a given problem.

Iowa Student Standards

Science

Grade **3** - Adopted: **2015**

STRAND / COURSE	IA.3-PS2.	Motion and Stability: Forces and Interactions
ESSENTIAL CONCEPT AND/OR SKILL		Students who demonstrate understanding can:
DETAILED DESCRIPTOR	3-PS2-2.	Make observations and/or measurements of an object’s motion to provide evidence that a pattern can be used to predict future motion.
STRAND / COURSE	IA.3-5-ETS1.	Engineering Design
ESSENTIAL CONCEPT AND/OR SKILL		Students who demonstrate understanding can:
DETAILED DESCRIPTOR	3-5-ETS1-1.	Define a simple design problem reflecting a need or a want that includes specified criteria for success and constraints on materials, time, or cost.
DETAILED DESCRIPTOR	3-5-ETS1-2.	Generate and compare multiple possible solutions to a problem based on how well each is likely to meet the criteria and constraints of the problem.
DETAILED DESCRIPTOR	3-5-ETS1-3.	Plan and carry out fair tests in which variables are controlled and failure points are considered to identify aspects of a model or prototype that can be improved.

Iowa Student Standards

Science

Grade **4** - Adopted: **2015**

STRAND / COURSE	IA.3-5-ETS1.	Engineering Design
ESSENTIAL CONCEPT AND/OR SKILL		Students who demonstrate understanding can:
DETAILED DESCRIPTOR	3-5-ETS1-1.	Define a simple design problem reflecting a need or a want that includes specified criteria for success and constraints on materials, time, or cost.
DETAILED DESCRIPTOR	3-5-ETS1-2.	Generate and compare multiple possible solutions to a problem based on how well each is likely to meet the criteria and constraints of the problem.
DETAILED DESCRIPTOR	3-5-ETS1-3.	Plan and carry out fair tests in which variables are controlled and failure points are considered to identify aspects of a model or prototype that can be improved.

Iowa Student Standards

Science

Grade **5** - Adopted: **2015**

STRAND / COURSE	IA.3-5-ETS1.	Engineering Design
ESSENTIAL CONCEPT AND/OR SKILL		Students who demonstrate understanding can:
DETAILED DESCRIPTOR	3-5-ETS1-1.	Define a simple design problem reflecting a need or a want that includes specified criteria for success and constraints on materials, time, or cost.
DETAILED DESCRIPTOR	3-5-ETS1-2.	Generate and compare multiple possible solutions to a problem based on how well each is likely to meet the criteria and constraints of the problem.
DETAILED DESCRIPTOR	3-5-ETS1-3.	Plan and carry out fair tests in which variables are controlled and failure points are considered to identify aspects of a model or prototype that can be improved.

Iowa Student Standards

Technology Education

Grade **K** - Adopted: **2018**

STRAND / COURSE		CSTA K-12 Computer Science Standards
ESSENTIAL CONCEPT AND/OR SKILL	CSTA.1A.	Level 1A (Ages 5-7)
DETAILED DESCRIPTOR	1A-CS.	Computing Systems
GRADE LEVEL EXPECTATION		Devices
EXAMPLE	1A-CS-01.	Select and operate appropriate software to perform a variety of tasks, and recognize that users have different needs and preferences for the technology they use. (P1.1)
STRAND / COURSE		CSTA K-12 Computer Science Standards
ESSENTIAL CONCEPT AND/OR SKILL	CSTA.1A.	Level 1A (Ages 5-7)
DETAILED DESCRIPTOR	1A-AP.	Algorithms & Programming
GRADE LEVEL EXPECTATION		Control
EXAMPLE	1A-AP-10.	Develop programs with sequences and simple loops, to express ideas or address a problem. (P5.2)
STRAND / COURSE		CSTA K-12 Computer Science Standards
ESSENTIAL CONCEPT AND/OR SKILL	CSTA.1A.	Level 1A (Ages 5-7)
DETAILED DESCRIPTOR	1A-AP.	Algorithms & Programming
GRADE LEVEL EXPECTATION		Program Development
EXAMPLE	1A-AP-12.	Develop plans that describe a program’s sequence of events, goals, and expected outcomes. (P5.1, P7.2)
EXAMPLE	1A-AP-15.	Using correct terminology, describe steps taken and choices made during the iterative process of program development. (P7.2)

Iowa Student Standards

Technology Education

Grade **1** - Adopted: **2018**

STRAND / COURSE		CSTA K-12 Computer Science Standards
ESSENTIAL CONCEPT AND/OR SKILL	CSTA.1A.	Level 1A (Ages 5-7)
DETAILED DESCRIPTOR	1A-CS.	Computing Systems
GRADE LEVEL		Devices

EXPECTATION		
EXAMPLE	1A-CS-01.	Select and operate appropriate software to perform a variety of tasks, and recognize that users have different needs and preferences for the technology they use. (P1.1)
STRAND / COURSE		CSTA K-12 Computer Science Standards
ESSENTIAL CONCEPT AND/OR SKILL	CSTA.1A.	Level 1A (Ages 5-7)
DETAILED DESCRIPTOR	1A-AP.	Algorithms & Programming
GRADE LEVEL EXPECTATION		Control
EXAMPLE	1A-AP-10.	Develop programs with sequences and simple loops, to express ideas or address a problem. (P5.2)
STRAND / COURSE		CSTA K-12 Computer Science Standards
ESSENTIAL CONCEPT AND/OR SKILL	CSTA.1A.	Level 1A (Ages 5-7)
DETAILED DESCRIPTOR	1A-AP.	Algorithms & Programming
GRADE LEVEL EXPECTATION		Program Development
EXAMPLE	1A-AP-12.	Develop plans that describe a program's sequence of events, goals, and expected outcomes. (P5.1, P7.2)
EXAMPLE	1A-AP-15.	Using correct terminology, describe steps taken and choices made during the iterative process of program development. (P7.2)

Iowa Student Standards

Technology Education

Grade **2** - Adopted: **2018**

STRAND / COURSE		CSTA K-12 Computer Science Standards
ESSENTIAL CONCEPT AND/OR SKILL	CSTA.1A.	Level 1A (Ages 5-7)
DETAILED DESCRIPTOR	1A-CS.	Computing Systems
GRADE LEVEL EXPECTATION		Devices
EXAMPLE	1A-CS-01.	Select and operate appropriate software to perform a variety of tasks, and recognize that users have different needs and preferences for the technology they use. (P1.1)
STRAND / COURSE		CSTA K-12 Computer Science Standards
ESSENTIAL CONCEPT AND/OR SKILL	CSTA.1A.	Level 1A (Ages 5-7)
DETAILED DESCRIPTOR	1A-AP.	Algorithms & Programming
GRADE LEVEL EXPECTATION		Control
EXAMPLE	1A-AP-10.	Develop programs with sequences and simple loops, to express ideas or address a problem. (P5.2)
STRAND / COURSE		CSTA K-12 Computer Science Standards
ESSENTIAL CONCEPT AND/OR SKILL	CSTA.1A.	Level 1A (Ages 5-7)
DETAILED DESCRIPTOR	1A-AP.	Algorithms & Programming
GRADE LEVEL EXPECTATION		Program Development
EXAMPLE	1A-AP-12.	Develop plans that describe a program's sequence of events, goals, and expected outcomes. (P5.1, P7.2)

EXAMPLE	1A-AP-15.	Using correct terminology, describe steps taken and choices made during the iterative process of program development. (P7.2)

Iowa Student Standards

Technology Education

Grade **3** - Adopted: **2018**

STRAND / COURSE		CSTA K-12 Computer Science Standards
ESSENTIAL CONCEPT AND/OR SKILL	CSTA.1B.	Level 1B (Ages 8-11)
DETAILED DESCRIPTOR	1B-AP.	Algorithms & Programming
GRADE LEVEL EXPECTATION		Variables
EXAMPLE	1B-AP-09.	Create programs that use variables to store and modify data. (P5.2)

STRAND / COURSE		CSTA K-12 Computer Science Standards
ESSENTIAL CONCEPT AND/OR SKILL	CSTA.1B.	Level 1B (Ages 8-11)
DETAILED DESCRIPTOR	1B-AP.	Algorithms & Programming
GRADE LEVEL EXPECTATION		Control
EXAMPLE	1B-AP-10.	Create programs that include sequences, events, loops, and conditionals. (P5.2)

STRAND / COURSE		CSTA K-12 Computer Science Standards
ESSENTIAL CONCEPT AND/OR SKILL	CSTA.1B.	Level 1B (Ages 8-11)
DETAILED DESCRIPTOR	1B-AP.	Algorithms & Programming
GRADE LEVEL EXPECTATION		Program Development
EXAMPLE	1B-AP-13.	Use an iterative process to plan the development of a program by including others'' perspectives and considering user preferences. (P1.1, P5.1)

Iowa Student Standards

Technology Education

Grade **4** - Adopted: **2018**

STRAND / COURSE		CSTA K-12 Computer Science Standards
ESSENTIAL CONCEPT AND/OR SKILL	CSTA.1B.	Level 1B (Ages 8-11)
DETAILED DESCRIPTOR	1B-AP.	Algorithms & Programming
GRADE LEVEL EXPECTATION		Variables
EXAMPLE	1B-AP-09.	Create programs that use variables to store and modify data. (P5.2)

STRAND / COURSE		CSTA K-12 Computer Science Standards
ESSENTIAL CONCEPT AND/OR SKILL	CSTA.1B.	Level 1B (Ages 8-11)
DETAILED DESCRIPTOR	1B-AP.	Algorithms & Programming
GRADE LEVEL EXPECTATION		Control

EXAMPLE	1B-AP-10.	Create programs that include sequences, events, loops, and conditionals. (P5.2)

STRAND / COURSE		CSTA K-12 Computer Science Standards
ESSENTIAL CONCEPT AND/OR SKILL	CSTA.1B.	Level 1B (Ages 8-11)
DETAILED DESCRIPTOR	1B-AP.	Algorithms & Programming
GRADE LEVEL EXPECTATION		Program Development
EXAMPLE	1B-AP-13.	Use an iterative process to plan the development of a program by including others'' perspectives and considering user preferences. (P1.1, P5.1)

Iowa Student Standards

Technology Education

Grade **5** - Adopted: **2018**

STRAND / COURSE		CSTA K-12 Computer Science Standards
ESSENTIAL CONCEPT AND/OR SKILL	CSTA.1B.	Level 1B (Ages 8-11)
DETAILED DESCRIPTOR	1B-AP.	Algorithms & Programming
GRADE LEVEL EXPECTATION		Variables
EXAMPLE	1B-AP-09.	Create programs that use variables to store and modify data. (P5.2)

STRAND / COURSE		CSTA K-12 Computer Science Standards
ESSENTIAL CONCEPT AND/OR SKILL	CSTA.1B.	Level 1B (Ages 8-11)
DETAILED DESCRIPTOR	1B-AP.	Algorithms & Programming
GRADE LEVEL EXPECTATION		Control
EXAMPLE	1B-AP-10.	Create programs that include sequences, events, loops, and conditionals. (P5.2)

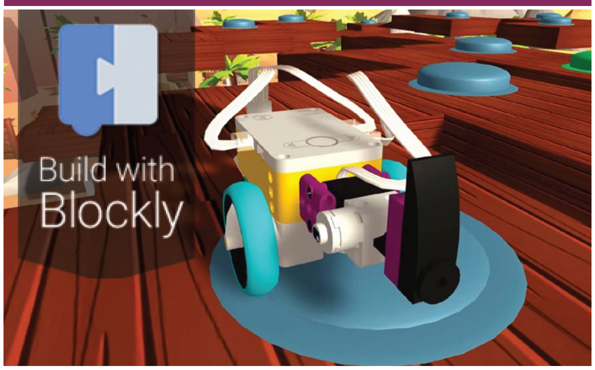
STRAND / COURSE		CSTA K-12 Computer Science Standards
ESSENTIAL CONCEPT AND/OR SKILL	CSTA.1B.	Level 1B (Ages 8-11)
DETAILED DESCRIPTOR	1B-AP.	Algorithms & Programming
GRADE LEVEL EXPECTATION		Program Development
EXAMPLE	1B-AP-13.	Use an iterative process to plan the development of a program by including others'' perspectives and considering user preferences. (P1.1, P5.1)

CoderZ™ Content Framework

CoderZ is an engaging self-paced virtual learning platform that provides interactive lessons and activities designed to teach students principles of coding within the exciting and challenging context of robotics.

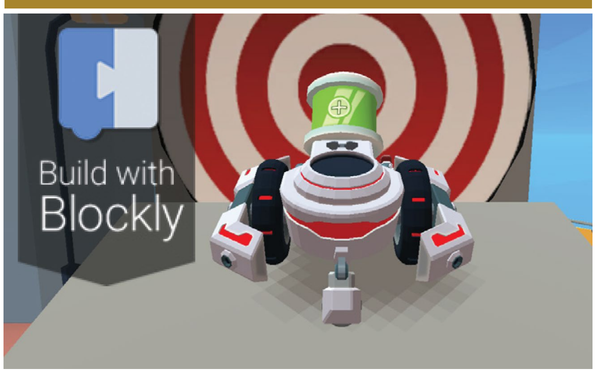
There are four courses available from which schools/districts may select to implement:

The CoderZ Adventure (Novice)

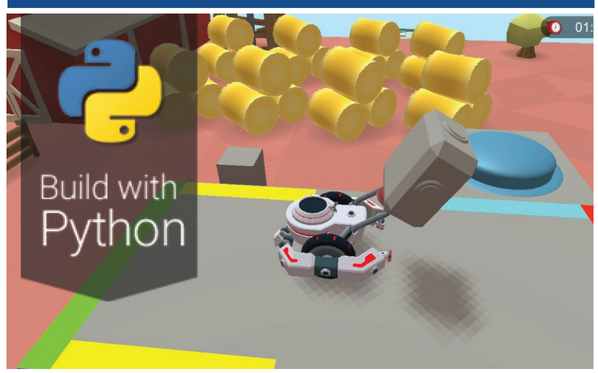


Cyber Robotics 101 (Beginner)

Cyber Robotics 102 (Intermediate)



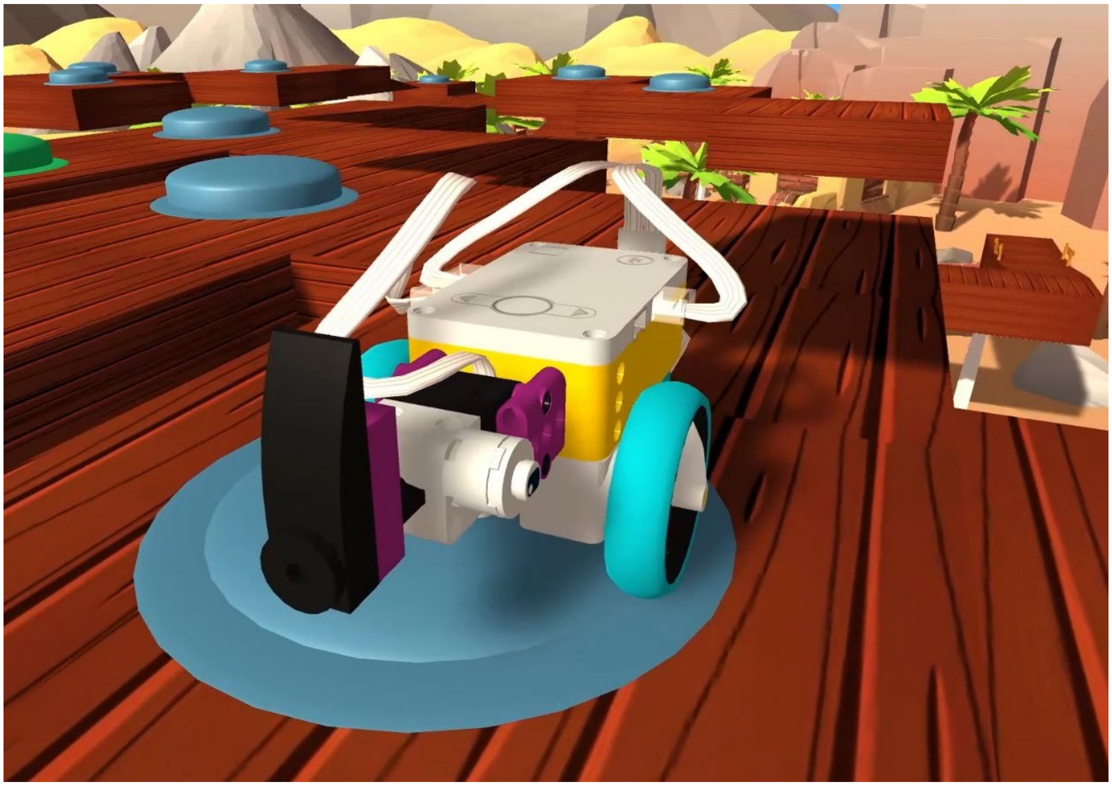
Python Gym (Advanced)



The CoderZ Content Framework provides an overview of the standards, skills, and concepts taught and supported within the learning platform, by the available CoderZ teacher resources - including the teacher/course guides, and with the integration of teacher lead lessons and activities as supported by CoderZ's professional development. Each course's content is individually framed for efficient referencing.

TABLE OF CONTENT

The CoderZ Adventure



In this introductory coding course students are introduced to the basic principles of coding within the context of robotics. While participating in a series of increasingly complex tasks students learn about the relationship between hardware and software, the role of mathematical modeling and data in computational problem solving, how basic programming constructs can be combined to create complex algorithms, as well as develop a foundation in key programming practices.

Standards, Skills, and Concepts

CSTA

Computer Systems

1B-CS-01	Describe how internal and external parts of computing devices function to form a system.
1B-CS-02	Model how computer hardware and software work together as a system to accomplish tasks.

Data and Analysis

1B-DA-07	Use data to highlight or propose cause-and-effect relationships, predicate outcomes, or communicate an idea.

Algorithms and Programming

1B-AP-08	Compare and refine multiple algorithms for the same task and determine which is the most appropriate
1B-AP-10	Create programs that include sequences, events, loops, and conditionals.
1B-AP-11	Decompose (break down) problems into smaller, manageable subproblems to facilitate the program development process.
1B-AP-13	Use an interactive process to plan the development of a program by including others' perspectives and considering user preferences.
1B-AP-15	Test and debug (identify and fix errors) in a program or algorithm to ensure it runs as intended.

Impacts of Computing

1B-IC-18	Discuss computing technologies that have changed the world, and express how those technologies influence, and are influenced by, cultural practices.

NGSS

3-5-ETS1-1	Define a simple design problem reflecting a need or a want that includes specified criteria for success and constraints on materials, time, or cost.
3-5-ETS1-2	Generate and compare multiple possible solutions to a problem based on how well each is likely to meet the criteria and constraints of the problem.
3-5-ETS1-37	Plan and carry out fair tests in which variables are controlled and failure points are considered to identify aspects of a model or prototype that can be improved.

K-12 Computer Science Framework

Computing Systems

- Troubleshooting - Check connections and power to resolve common issues. Explain and demonstrate how rebooting a machine is commonly an effective strategy.

Algorithms and Programming

- Algorithms - Compare algorithms and select the one most appropriate for a specific context/task.
- Control - Recognize patterns and effectively use loops to enhance the efficiency of writing code.
- Program Development - Use an iterative process involving design, implementation, and review when developing code.

Impacts of Computing

- Social Interaction - Analyze how computing technology's facilitation of communication and innovations influences social institutions such as family, education, and the economy.

CC Math

Mathematical Practice

MP1 Make sense of problems and persevere in solving them.

MP2 Reason abstractly and quantitatively.

MP3 Construct viable arguments and critique the reasoning of others.

MP4 Model with mathematics.

MP5 Use appropriate tools strategically.

MP6 Attend to precision.

MP7 Look for and make use of structure.

MP8 Look for and express regularity in repeated reasoning.

21st Century Skills

- Creativity and Innovation
- Critical Thinking and Problem Solving
- Communication
- Collaboration

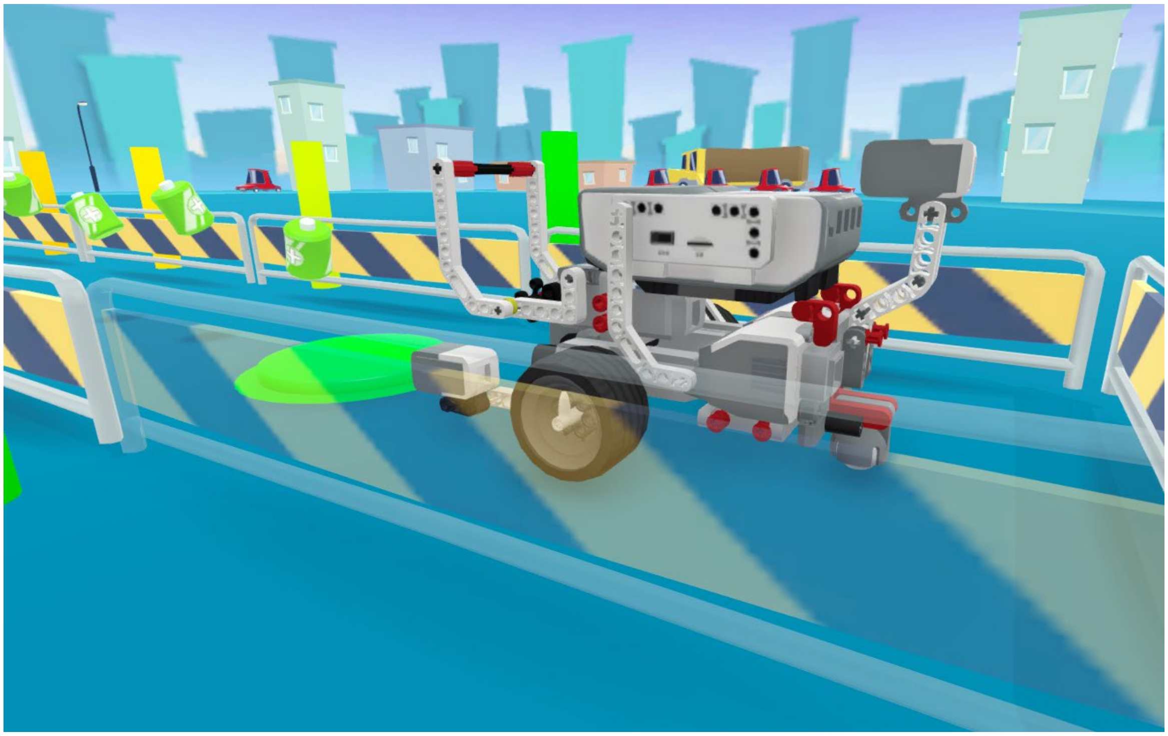
Computational Thinking Skills

- Decomposition
- Pattern Recognition
- Algorithm Design

Robotics Concepts and Skills

- Motion Planning - Direction

Cyber Robotics 101



In this beginner's coding course students are introduced to the principles of coding within the context of robotics. While engaged in a series of activities with increasing complexity students learn the essential principles of robotics including motion planning, motion planning with sensors, the relationship between hardware and software, the role of mathematical modeling and data in computational problem solving, the importance of quality planning, as well as develop the ability to execute good programming practices. CoderZ and supporting integrated content is sequenced to meet STEAM objectives and is designed to provide students authentic learning opportunities to increase interest and engagement while moving from acquisition of knowledge to transference of learning.

Standards, Skills, and Concepts

CSTA

Computer Systems

2-CS-02	Design projects that combine hardware and software components to collect and exchange data.

Algorithms and Programming

2-AP-10	Use flowcharts and/or pseudocode to address complex problems as algorithms.
2-AP-12	Design and iteratively develop programs that combine control structures, including nested loops and compound conditionals.
2-AP-13	Decompose problems and subproblems into parts to facilitate the design, implementation, and review of programs.
2-AP-14	Create procedures with parameters to organize code and make it easier to use.
2-AP-17	Systematically test and refine programs using a range of test cases.
2-AP-18	Distribute tasks and maintain a project timeline when collaboratively developing computational artifacts.
2-AP-19	Document programs in order to make them easier to follow, test, and debug.

Impacts of Computing

2-IC-20	Compare tradeoffs associated with computing technologies that affect people's everyday activities and career options.
2-IC-21	Discuss issues of bias and accessibility in the design of existing technologies.
2-IC-23	Describe tradeoffs between allowing information to be public and keeping information private and secure.

NGSS

MS-ETS1-2	Evaluate competing design solutions using a systematic process to determine how well they meet the criteria and constraints of the problem.
MS-ETS1-3	Analyze data from tests to determine similarities and differences among several design solutions to identify the best characteristics of each that can be combined into a new solution to better meet the criteria for success.

K-12 Computer Science Framework

Computing Systems

- Devices - Examine the interaction between humans and computing devices. Evaluate and analyze the advantages, disadvantages, and recognize the unintended consequences.
- Troubleshooting - Deploy a structured process to troubleshoot problems within a system to ensure potential solutions are not overlooked and simple issues are resolved.

Algorithms and Programming

- Algorithms - Design, test, and debug algorithms that are readable and easy to follow.
- Variables - Understand and use variables to represent and process data to produce varying outputs.
- Control - Use loops, conditions, and other control structures to create more complex programs.
- Modularity - Use functions to make code easier to reuse and read.

Networks and the Internet

- Cybersecurity - Explain the importance of protecting information sent and received across networks from unauthorized access and modification (encryption and Hypertext Transfer Protocol Secure - HTTPS).

Data and Analysis

- Collection - Discuss how using sensors with a robot to collect data regarding its environment and make decisions about the next steps to take is similar to the human process and how robots and humans interact.

CC Math

Mathematical Practice

MP1 Make sense of problems and persevere in solving them.

MP2 Reason abstractly and quantitatively.

MP3 Construct viable arguments and critique the reasoning of others.

MP4 Model with mathematics.

MP5 Use appropriate tools strategically.

MP6 Attend to precision.

MP7 Look for and make use of structure.

MP8 Look for and express regularity in repeated reasoning.

21st Century Skills

Learning and Innovation Skills

- Creativity and Innovation
- Critical Thinking and Problem Solving
- Communication and Collaboration

Information, Media and Technology Skills

- Information Literacy
- Information and Communications Technology (ICT)

Life and Career Skills

- Flexibility and Adaptability
- Initiative and Self-direction
- Social and Cross-Cultural Skills
- Productivity and Accountability
- Leadership and Responsibility

Computational Thinking Skills

- Decomposition
- Pattern Recognition
- Algorithm Design
- Abstraction

Robotics Concepts and Skills

- Motion Planning
  - Direction
  - Power/Speed
  - Distance
  - Duration
  - Arm
- Sensor-based Motion Planning
- Motors and Optical Encoders
- Sensors and Controllers
  - Touch - collision
  - Gyroscope - rotation
  - Ultrasonic - distance
  - Light/Color - surface color/brightness

Cyber Robotics 102

Cyber Robotics 102 (CR102) is a continuation of the Cyber Robotics 101 course. This course teaches STEM and coding topics using gamification in a physics-based environment. Through a series of activities with increasing complexity CR102 introduces autonomous systems, teaches scanning and mapping an environment, error correction methods, and different system control algorithms. Cyber Robotics 102 provides a deeper coding experience with more opportunities to use conditionals, variables, functions, etc. CoderZ and supporting integrated content is sequenced to meet STEAM objectives and is designed to provide students authentic learning opportunities to increase interest and engagement while moving from acquisition of knowledge to transference of learning. By the end of the course students will understand the forces of physics acting on robots and the influence those forces have on a robot's performance. Students will effectively control and program a robot to interact with objects around it and safely navigate through changing environments.

Standards, Skills, and Concepts

CSTA

Computer Systems

2-CS-02	Design projects that combine hardware and software components to collect and exchange data.

Algorithms and Programming

2-AP-10	Use flowcharts and/or pseudocode to address complex problems as algorithms.
2-AP-11	Create clearly named variables that represent different data types and perform operations on their values.
2-AP-12	Design and iteratively develop programs that combine control structures, including nested loops and compound conditionals.
2-AP-13	Decompose problems and subproblems into parts to facilitate the design, implementation, and review of programs.
2-AP-14	Create procedures with parameters to organize code and make it easier to use.
2-AP-17	Systematically test and refine programs using a range of test cases.
2-AP-18	Distribute tasks and maintain a project timeline when collaboratively developing computational artifacts.
2-AP-19	Document programs in order to make them easier to follow, test, and debug.

Impacts of Computing

2-IC-20	Compare tradeoffs associated with computing technologies that affect people's everyday activities and career options.
2-IC-21	Discuss issues of bias and accessibility in the design of existing technologies.
2-IC-23	Describe tradeoffs between allowing information to be public and keeping information private and secure.

NGSS

MS-ETS1-2	Evaluate competing design solutions using a systematic process to determine how well they meet the criteria and constraints of the problem.
MS-ETS1-3	Analyze data from tests to determine similarities and differences among several design solutions to identify the best characteristics of each that can be combined into a new solution to better meet the criteria for success.

K-12 Computer Science Framework

Computing Systems

- Devices - Examine the interaction between humans and computing devices. Evaluate and analyze the advantages, disadvantages, and recognize the unintended consequences.
- Troubleshooting - Deploy a structured process to troubleshoot problems within the system to ensure potential solutions are not overlooked and simple issues are resolved.
- Algorithms and Programming
- Algorithms - Design, test, and debug algorithms that are readable and easy to follow.
- Variables - Understand and use variables to represent and process data to produce varying outputs.
- Control - Use loops, conditions, and other control structures to create more complex programs.
- Modularity - Use functions to make code easier to reuse and read.

Networks and the Internet

- Cybersecurity - Explain the importance of protecting information sent and received across networks from unauthorized access and modification (encryption and Hypertext Transfer Protocol Secure - HTTPS).

Data and Analysis

- Collection - Discuss how using sensors with a robot to collect data regarding its environment and make decisions about the next steps to take is similar to the human process and how robots and humans interact.

CC Math

Mathematical Practice

MP1 Make sense of problems and persevere in solving them.

MP2 Reason abstractly and quantitatively.

MP3 Construct viable arguments and critique the reasoning of others.

MP4 Model with mathematics.

MP5 Use appropriate tools strategically.

MP6 Attend to precision.

MP7 Look for and make use of structure.

MP8 Look for and express regularity in repeated reasoning.

21st Century Skills

Learning and Innovation Skills

- Creativity and Innovation
- Critical Thinking and Problem Solving
- Communication and Collaboration

Information, Media and Technology Skills

- Information Literacy
- Information and Communications Technology (ICT)

Life and Career Skills

- Flexibility and Adaptability
- Initiative and Self-direction
- Social and Cross-Cultural Skills
- Productivity and Accountability
- Leadership and Responsibility

Computational Thinking Skills

- Decomposition
- Pattern Recognition
- Algorithm Design
- Abstraction

Robotics Concepts and Skills

- Motion Planning
    - Direction
    - Power/Speed
    - Distance
    - Duration
    - Brake
    - Arms
- Sensor-based Motion Planning
- Motors and Optical Encoders
- Sensors and Controllers
    - Touch - collision
    - Gyroscope - rotation and tilts (x, y, z)
    - Ultrasonic - distance
    - Light/Color - surface color/brightness

Python Gym

Python Gym is an advanced course providing teachers and students the opportunity to code in Python, a text-based computer language, using the CoderZ physically-based environment introduced in Cyber Robotics 102. While engaged in a series of activities with increasing complexity students learn the essential principles of text-based coding including syntax and debugging and continue to develop best programming practices for robotics including; motion planning, motion planning with sensors, physics, computational problem solving, and the importance of quality planning. Python Gym provides opportunities to use conditionals, variables, functions, modules and more.

CoderZ and supporting integrated content is sequenced to meet STEAM objectives and is designed to provide students authentic learning opportunities to increase interest and engagement while moving from acquisition of knowledge to transference of learning.

Standards, Skills, and Concepts

CSTA

Computer Systems

2-CS-02	Design projects that combine hardware and software components to collect and exchange data.
3A-CS-01	Explain how abstractions hide the underlying implementation details of computing systems embedded in everyday objects.

Algorithms and Programming

2-AP-10	Use flowcharts and/or pseudocode to address complex problems as algorithms.
2-AP-11	Create clearly named variables that represent different data types and perform operations on their values.
2-AP-12	Design and iteratively develop programs that combine control structures, including nested loops and compound conditionals.
2-AP-13	Decompose problems and subproblems into parts to facilitate the design, implementation, and review of programs.
2-AP-14	Create procedures with parameters to organize code and make it easier to use.
2-AP-17	Systematically test and refine programs using a range of test cases.
2-AP-18	Distribute tasks and maintain a project timeline when collaboratively developing computational artifacts.
2-AP-19	Document programs in order to make them easier to follow, test, and debug.

3A-AP-17	Decompose problems into smaller components through systematic analysis, using constructs such as procedures, modules, and/or objects.
3A-AP-18	Create artifacts by using procedures within a program, combinations of data and procedures, or independent but interrelated programs.
3B-AP-11	Evaluate algorithms in terms of their efficiency, correctness, and clarity.
3B-AP-16	Demonstrate code reuse by creating programming solutions using libraries and APIs.
3B-AP-23	Evaluate key qualities of a program through a process such as code review.

Impacts of Computing

2-IC-20	Compare tradeoffs associated with computing technologies that affect people's everyday activities and career options.
2-IC-21	Discuss issues of bias and accessibility in the design of existing technologies.
2-IC-23	Describe tradeoffs between allowing information to be public and keeping information private and secure.
3A-IC-24	Evaluate the ways computing impacts personal, ethical, social, economic, and cultural practices.
3A-IC-28	Explain the beneficial and harmful effects that intellectual property laws can have on innovation.
3A-IC-30	Evaluate the social and economic implications of privacy in the context of safety, law, or ethics.
3B-IC-27	Predict how computational innovations that have revolutionized aspects of our culture might evolve.

NGSS

MS-ETS1-2	Evaluate competing design solutions using a systematic process to determine how well they meet the criteria and constraints of the problem.
MS-ETS1-3	Analyze data from tests to determine similarities and differences among several design solutions to identify the best characteristics of each that can be combined into a new solution to better meet the criteria for success.

K-12 Computer Science Framework

Computing Systems

- Devices – Examine
- e the interaction between humans and computing devices. Evaluate and analyze the advantages, disadvantages, and recognize the unintended consequences.
- Troubleshooting - Deploy a structured process to troubleshoot problems within the system to ensure potential solutions are not overlooked and simple issues are resolved.
- Hardware and Software - Explore the interactions between hardware, software and user of a computing system and evaluate the flow of information.

Algorithms and Programming

- Algorithms - Design, test, and debug algorithms that are readable and easy to follow. Evaluate and select algorithms based on performance, reusability, and ease of implementation.

- Variables - Understand and use variables to represent and process data to produce varying outputs.
  Use data structures to effectively manage program complexity (based on functionality, storage, and performance tradeoffs).

- Control - Use loops, conditions, and other control structures to create more complex programs.
  Evaluate control structures and control structure combinations considering the tradeoffs related to implementation, readability, and program performance.

- Modularity - Use functions to make code easier to reuse and read.
  Use modules to develop and manage complex tasks/programs.

Networks and the Internet

- Cybersecurity - Explain the importance of protecting information sent and received across networks from unauthorized access and modification (encryption and Hypertext Transfer Protocol Secure - HTTPS).

Data and Analysis

- Collection - Discuss how using sensors with a robot to collect data regarding its environment and make decisions about the next steps to take is similar to the human process and how robots and humans interact.

CC Math

Mathematical Practice

MP1 Make sense of problems and persevere in solving them.

MP2 Reason abstractly and quantitatively.

MP3 Construct viable arguments and critique the reasoning of others.

MP4 Model with mathematics.

MP5 Use appropriate tools strategically.

MP6 Attend to precision.

MP7 Look for and make use of structure.

MP8 Look for and express regularity in repeated reasoning.

21st Century Skills

Learning and Innovation Skills

- Creativity and Innovation
- Critical Thinking and Problem Solving
- Communication and Collaboration

Information, Media and Technology Skills

- Information Literacy
- Information and Communications Technology (ICT)

Life and Career Skills

- Flexibility and Adaptability
- Initiative and Self-direction
- Social and Cross-Cultural Skills
- Productivity and Accountability
- Leadership and Responsibility

Computational Thinking Skills

- Decomposition
- Pattern Recognition
- Algorithm Design
- Abstraction

Robotics Concepts and Skills

- Motion Planning
  - Direction
  - Power/Speed
  - Distance
  - Duration
  - Brake
  - Arms
- Sensor-based Motion Planning
- Motors and Optical Encoders
- Sensors and Controllers
  - Touch - collision
  - Gyroscope - rotation and tilts (x, y, z)
  - Ultrasonic - distance
  - Light/Color - surface color/brightness

Exhibit 15 – Standards

- Include a list of standards addressed in the curricular materials
  - **CoderZ Adventure** (see attached PDF) demonstrating alignment to K-5
  - **Cyber Robotics 101** (see attached PDF) demonstrating alignment to 6-8
  - **Cyber Robotics 102** (see attached PDF) demonstrating alignment to 6-12
  - **Python Gym** (see attached PDF) demonstrating alignment to 9-12
- Describe how standards are age and academically appropriate
  - CoderZ curriculum is aligned with standards and leveled appropriately for student experience with coding and robotics.
  - Each lesson in the CoderZ curriculum begins with a definition of the **student learning outcomes,** clearly indicating the knowledge, skills and abilities individual students should possess and demonstrate upon completion of a learning experience or sequence of learning experiences.
  - CoderZ courses allow for multiple entry points to ensure students have any required prior knowledge for that course and students are always working with content that focuses on skills and uses high-quality 3-D graphics to engage all students.
- Describe how content is aligned to standards
  - CoderZ curriculum partners with **EdGate Correlation Services** to ensure and accurate and neutral perspective of alignment.
  - Through EdGate CoderZ alignment may be done by standard or by course.
  - **CoderZ Content Framework** (also attached as a separate PDF for Exhibit 15)
- Provide a detailed description of how three standards are met
  - **Iowa Student Standards Technology Education Grade 1 - Adopted: 2018**
    STRAND / COURSE CSTA K-12 Computer Science Standards
    ESSENTIAL CONCEPT AND/OR SKILL CSTA.1A. Level 1A (Ages 5-7)
    DETAILED DESCRIPTOR 1A-AP. Algorithms & Programming
    GRADE LEVEL EXPECTATION Control
    EXAMPLE 1A-AP10. Develop programs with sequences and simple loops, to express ideas or address a problem. (P5.2)
    CoderZ Adventure > The Lost City > Mission 6: Path Maker
    To successfully solve the mission students are required to:
    - Decompose the simulation and plan the path of the robot (sequencing)
    - Measure distance using measuring tools within the simulation plane
    - Recognize patterns
    - Create an algorithm with appropriate use of a loop
  - **Iowa Student Standards Technology Education Grade 6 - Adopted: 2018**
    STRAND / COURSE CSTA K-12 Computer Science Standards
    ESSENTIAL CONCEPT AND/OR SKILL CSTA.2. Level 2 (Ages 11-14)
    DETAILED DESCRIPTOR 2-AP. Algorithms & Programming
    GRADE LEVEL EXPECTATION Control
    EXAMPLE 2-AP-12. Design and iteratively develop programs that combine control structures, including nested loops and compound conditionals. (P5.1, P5.2)

CoderZ Cyber Robotics 101 > Challenge Mission III > Conclusion Challenge: Part 2
To successfully solve the mission students are required to:

- Decompose the simulaiton and plan the path of the robot (sequencing)
- Create an algorithm with the appropriate combination of control structures
    - Wait until (control with condition)
    - Repeat until loop (conditional)
    - Nested If/Else loop (compound conditional)
- This challenge contains stationary and moving obstacles which requires students to use sensors and chunk code, test, write the next iteration and repeat the process until the robot successfully navigates its environment. (iteration)

- **Iowa Student Standards Technology Education Grade 9 - Adopted: 2018**
  STRAND / COURSE CSTA K-12 Computer Science Standards
  ESSENTIAL CONCEPT AND/OR SKILL CSTA.3A. Level 3A (Ages 14 -16)
  DETAILED DESCRIPTOR 3A-AP. Algorithms & Programming
  GRADE LEVEL EXPECTATION Control EXAMPLE 3A-AP15. Justify the selection of specific control structures when tradeoffs involve implementation, readability, and program performance, and explain the benefits and drawbacks of choices made. (P5.2)
  CoderZ Cyber Robotics 102 > A Hard Block Life > All Missions
  Students use encoders, gyro and reflection values, and best coding practices learned to successfully solve missions. Students must:
    - Decompose the simulation
    - Determine information to be collected and the input tools (sensors, encoders) required
    - Select appropriate controls to build the robots decision-making capabilities
    - Evaluate their final iteration based on readability, and program performance, and explain the benefits and drawbacks of choices made. If better iteration is possible teachers provide guidance and redirect students to try again.

EXHIBIT 16 – Evidence of Effectiveness

Provide how evaluation data was collected and an example of how an assessment informed program practice.

The CoderZ CSTEM focus for virtual coding and robotics to reach all learners is based on engaging students with 3-D graphics games. Data is entirely based on teacher and student input to show program practice results. Please preview this teacher video and this administrator video for evidence.

Provide source(s) of evaluation evidence, including any third party, independent evaluation.

CoderZ source(s) of evaluation evidence are specifically based on teacher and student feedback.

Describe the benefits to students and educators in terms of content and skills growth, attitudes and intentions.

Student motivation is supported and shown in this video with a teacher implementing CoderZ.

Provide evidence of testing to show what measures were used to measure student learning, as a result of engaging with the curriculum.

CoderZ virtual robotics and coding brings math and science skills to life supporting student's success in tested areas. CoderZ measures student learning through engagement.

Provide evidence of engaging learners who traditionally have been underrepresented in computer science, including but not limited to English Learners, persons of low income (FRL) and students with disabilities, as well as engaging learners to address race-ethnicity and gender gaps.

CoderZ levels up hybrid learning in STEM for all students including traditionally underrepresented populations to remove perceived barriers for students and find the students who didn't know they would love coding and robotics. This November 2020 webinar shows how CoderZ opens the door for all students. And, this January 2021 webinar shares inclusiveness in STEM with two educators.

Provide evidence that participation in the curriculum resulted in positive learning outcomes for students.

Case Study: Central Jr. High School

Bryan Harston, J.D.
Engineering & Robotics Teacher
Central Jr. High School STEM Program
HEB Independent School District
bryanharston@hebisd.edu
(817) 354-3350 x7534

About Central Jr. High School

Part of the Hurst Euless Bedford Independent School District (Texas), Central Jr. High School is a relatively small school with a comparatively big heart. The 1,100+ students experience a range of learning and enrichment activities, from academic programs to fine arts to robotics.

Meeting the Challenge of COVID Head On.

The Challenge

Continuing in Context of COVID

Like millions of educators across the country, Bryan was thrust into the biggest challenge of his career thanks to COIVD-induced school closures. Ever the resourceful teacher, Bryan knew he had to maintain the continuity of learning despite COVID. His professionalism as an educator and his sense of personal responsibility to his students meant that he had to find a program that would continue their long-standing tradition of exceptional robotics instructions. All that said - Bryan had three classes of robotics students using Lego EV3s, but no way to continue building on all the progress they had made year to date. The district wasn't anywhere close to 1:1 computing at that time and, besides, any program offering a semblance of online robotics was either lacking in sophistication or was prohibitively expense. So, what were they to do?

CoderZ Just in Time

Through an outreach program by CoderZ to make its virtual robotics platform freely available to schools for a limited time, Bryan found just in CoderZ precisely the kind of platform he needed to maintain continuity of learning. It started with seeing Lego's EV3 robotic embedded in the platform's environment, and his interest expanded when Bryan saw the deep working relationship between CoderZ and Amazon Future Engineers, a philanthropic initiative to bolster STEM, coding, and robotics in education.

Because students were learning remotely, it was imperative that they had fast, easy access to a learning environment that could not only connect kids to content but could also connect them to each other. CoderZ, optimized for Chromebooks and readily available on any Internet browser, offers a series of curricula designed for students in upper elementary through high school grades. With a "low floor, high ceiling" approach, the platform reduces the barriers to entry for students with limited experience (low floor) while also offering a robust, challenging environment for students with a deeper skill set and experience (high ceiling).

Along the way, Bryan also came to understand the unintended upside to virtual robotics. Traditional robotics are valuable and will always have a place at the center of instruction. But they are also prone to decay and mechanical failure. These can lead to questions about the veracity of code versus the reliability of a robot in decline. The care and maintenance of robots is an important element of mechanical engineering but falls outside the scope of coding for robotics. And... they are hard to clean in a way that meets safety standards for instruction in context of COVID. Virtual robots, on the other hand, require no clearing, never breakdown, and always run true.

Looking Ahead: Engagement, No Matter What

Bryan's district re-entered school using hybrid instruction, which underscores the importance of easy, equitable access to a learning environment made for robotics. And along the way, engagement is key to Bryan and his fellow STEM teachers. As Bryan noted from the spring, "while having some kids access the classroom via Google did reduce in-class distractions, it was also easy for them to check out. We need a way to keep all kids engaged."

Bryan cited several reasons that his adoption of CoderZ has led to greater engagement, including the fact that it is "so visually engaging, so good, cleverly designed and realistic." He also cited the fact that being a virtual platform means that it is "web-based and Chromebook compatible, which instantly solves the problem for a huge percentage of students."

Bryan, and the millions of teachers looking for the best way to serve students in these trying times, are innovating daily to drive engagement and impact learning. His move to CoderZ meant that he and his students could move forward with confidence. Now that's meeting the challenge head on.

EXHIBIT 17 – Optional Features

As an optional feature for schools across Iowa, we invite educators and their students to engage in CoderZ League, a national, virtual coding competition entering its fourth year.

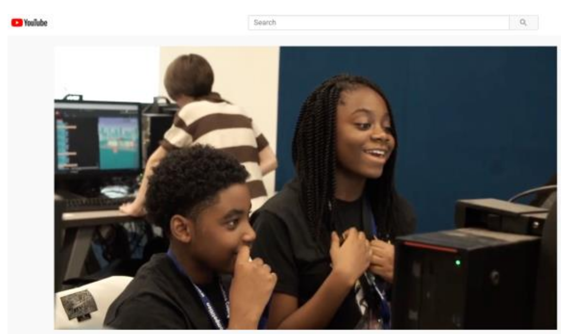
In collaboration with the Intelitek STEM & CTE Foundation, we have have been running a Cyber Robotics Coding Competition annually for students around the country. As we submit this RFP - we are launching our 4th annual competition, which we are excited to relaunch as CoderZ League.

This is a virtual competition that is designed with a "low-floor, high ceiling" framework that ensures all students feel welcome. One of the main goals of CoderZ League is centered on inclusivity, bringing coding and robotics to every student, breaking away from the mindset that such skills are confined only to clubs or extra-curricular activities.

CoderZ League offers competitive events for beginner level coders (Grades 5-8), as well as more advanced coders (Grades 7-12). Students work individually and in teams to solve challenges by writing code. The competition has succeeded to get students even more engaged in learning. Students come into the classroom and ask the instructor to continue working on the competition challenges – so they can improve the ranking of the school on the state leaderboard.

Over 100,000 students have competed in CoderZ League to date and the 2020-2021 competition, set to launch in August 2020, is shaping up to be as strong as ever. Nearly half of all participants are girls and more than one-third of participating schools are rural, both initiative of our tireless commitment to equity in STEM.

We invite you to review a video introduction together with teacher and student testimonials.



https://www.youtube.com/watch?v=AxTzDhurg5I&t=4s

Pricing for joining the Coderz League is designed to be affordable and accessible to all:

Participation Level Type	# of Seats	Price Cap Per School
Class/Club	30	$450.00
School	240	$1,500.00

"It's exciting to see kids so engaged in learning skills that are going to be important for their future. They're learning, but they're having fun at the same time."

Frank Edelblut
New Hampshire's education commissioner

Important to note: CoderZ curriculum supports CoderZ League beyond the competition while preparing students with skills for competition(s).

Exhibit 18 – Addendums

- Provide signed copy of posted RFP addendums (see attached)

February 15, 2021

To: All Potential Respondents
From: Kelli Sizenbach, Purchasing Agent
Subject: RFP1421282045

Addendum One

Please amend the subject RFP to include answers to the following timely received questions:

Q1. Will there be a need to crosswalk ISTE or other subject area standards to the Iowa CSTA standards?

A1. No. This is not required.

Q2. Are there metadata requirements for the curriculum and professional learning materials?

A2. No. There is no requirement.

Q3. Do you anticipate extending the bid due date?

A3. No.

Q4. What additional details are you willing to provide, if any, beyond what is stated in bid documents concerning how you will identify the winning bid?

A4. All information will be provided via the RFP document or an amendment.

Q5. Are Iowa school districts required to use certain LMSs, or do they choose their own? If there are preferred LMSs, could you provide a list?

A5. The decision about whether to use an LMS, and which LMS if any, is a local decision.

Q6. Section 1.12 of the "General Terms and Conditions for Service Contracts/Solicitations" specifies that all deliverables become the intellectual property of the State and Agency. Our company intends to offer a turn-key computer science curriculum solution that aligns precisely to Iowa standards, but we need to retain rights to our existing IP. Can you offer clarification about what type of deliverable the State would expect rights to?

A6. If you have proposed changes to the terms and conditions, please provide those as part of your response.

Q7. Are you looking for standalone programs that teach the Iowa/CSTA standards for each grade band? Meaning, curricula that can be used to teach "specials/exploratory" classes?

A7. We will consider both stand-alone curriculum and integrated curriculum.

Q8. If we have nothing available for the K-5 and 6-8 bands, but have items for the 9-12/AP band, are we allowed to submit them?

A8. Yes. Please note, separate proposals are required for each grade band.

Q9. Do programs need to be platform specific? PC/Apple?

A9. No.

Q10. Should we include samples. If so, who/where should we send it to?

A10. You are encouraged to provide sample artifacts from the curriculum and professional development. Samples should be included with your submission.

Q11. Will the State provide an invoice when districts choose our product?

A11. No. Individual districts will work directly with their selected vendor.

Please acknowledge receipt of this addendum by signing in the space provided below, and <u>return this letter with your offer (do not send back separately).</u>

I hereby acknowledge receipt of this addendum.

Signature ______________________ Date 2/26/2021

Ido Yerushalm
Typed or Printed Name

February 23, 2021

To: All Potential Respondents
From: Kelli Sizenbach, Purchasing Agent
Subject: RFP1421282045

Addendum Two

Please amend the subject RFP to include answers to the following timely received questions:

Q1. I see that in Exhibit 5 of this RFP, Letters of Reference are requested. Is the department looking specifically for letters that are written by our references, or only contact information? If actual letters are required, is there any kind of prompt that the references can follow? Would our references then send those directly to us to include in the RFP?

A1. Please provide letters of reference. There is not a specific template for these letters.

Q2. Would the resources on this list be the ONLY resources that districts could use the Computer Science Professional Development Incentive Funds on, or would the list be a starting point to help districts who don't know what their options are?

A2. The Computer Science Professional Development Incentive Fund cannot be used for curriculum. Recipients of Computer Science Professional Development Incentive fund grants choose their professional development approach. A different source of funding is available for districts and accredited nonpublic schools who are interested in the resources on our list.

Please acknowledge receipt of this addendum by signing in the space provided below, and <u>return this letter with your offer (do not send back separately).</u>

I hereby acknowledge receipt of this addendum.

Signature ______________________ Date 2/26/2021

Ido Yerushalmi
Typed or Printed Name

Attachment #3

Form 22 – Request for Confidentiality

SUBMISSION OF THIS FORM 22 IS REQUIRED

THIS FORM 22 (FORM) MUST BE COMPLETED AND INCLUDED WITH YOUR PROPOSAL. THIS FORM 22 IS REQUIRED WHETHER THE PROPOSAL DOES OR DOES NOT CONTAIN INFORMATION FOR WHICH CONFIDENTIAL TREATMENT WILL BE REQUESTED. FAILURE TO SUBMIT A COMPLETED FORM 22 WILL RESULT IN THE PROPOSAL TO BE CONSIDERED NON-RESPONSIVE AND ELIMINATED FROM EVALUATION. COMPLETE PART 1 OF THIS FORM IF NO INFORMATION PROPOSAL DOES NOT CONTAIN CONFIDENTIAL INFORMATION. COMPLETE PART 2 OF THIS FORM IF THE PROPOSAL DOES CONTAIN CONFIDENTIAL INFORMATION.

1. **Confidential Treatment Is Not Requested**
   A Respondent not requesting confidential treatment of information contained in its Proposal shall complete Part 1 of Form 22 and submit a signed Form 22 Part 1 with the Proposal.

2. **Confidential Treatment of Information is Requested**
   A Respondent requesting confidential treatment of specific information shall: (1) fully complete and sign Part 2 of Form 22, (2) conspicuously mark the outside of its Proposal as containing confidential information, (3) mark each page upon which the Respondent believes confidential information appears **and CLEARLY IDENTIFY EACH ITEM for which confidential treatment is requested; MARKING A PAGE IN THE PAGE MARGIN IS NOT SUFFICIENT IDENTIFICATION**, and (4) submit a "Public Copy" from which the confidential information has been excised.

   Form 22 will not be considered fully complete unless, for each confidentiality request, the Respondent: (1) enumerates the specific grounds in Iowa Code Chapter 22 or other applicable law that supports treatment of the information as confidential, (2) justifies why the information should be maintained in confidence, (3) explains why disclosure of the information would not be in the best interest of the public, and (4) sets forth the name, address, telephone, and e-mail for the person authorized by Respondent to respond to inquiries by the Agency concerning the confidential status of such information.

   The Public Copy from which confidential information has been excised is in addition to the number of copies requested in Section 3 of this RFP. The confidential information must be excised in such a way as to allow the public to determine the general nature of the information removed and to retain as much of the Proposal as possible.

   Failure to request information be treated as confidential as specified herein shall relieve Agency and State personnel from any responsibility for maintaining the information in confidence. Respondents may not request confidential treatment with respect to pricing information and transmittal letters. A Respondent's request for confidentiality that does not comply with this form or a Respondent's request for confidentiality on information or material that cannot be held in confidence as set forth herein are grounds for rejecting Respondent's Proposal as non-responsive. Requests to maintain an entire Proposal as confidential will be rejected as non-responsive.

   If Agency receives a request for information that Respondent has marked as confidential and if a judicial or administrative proceeding is initiated to compel the release of such information, Respondent shall, at its sole expense, appear in such action and defend its request for confidentiality. If Respondent fails to do so, Agency may release the information or material with or without providing advance notice to Respondent and with or without affording Respondent the opportunity to obtain an order restraining its release from a court possessing competent jurisdiction. Additionally, if Respondent fails to comply with the request process set forth herein, if Respondent's request for confidentiality is unreasonable, or if Respondent rescinds its request for confidential treatment, Agency may release such information or material with or without providing advance notice to Respondent and with or without affording Respondent the opportunity to obtain an order restraining its release from a court possessing competent jurisdiction.

Part 1 – No Confidential Information Provided

Confidential Treatment Is Not Requested

Respondent acknowledges that proposal response contains no confidential, secret, privileged, or proprietary information. There is no request for confidential treatment of information contained in this proposal response.

This Form must be signed by the individual who signed the Respondent's Proposal. The Respondent shall place this Form completed and signed in its Proposal.

- ***Fill in and sign the following if you have provided no confidential information. If signing this Part 1, do not complete Part 2.***

Intelitek Inc.	RFP 1421282045	COMPUTER SCIENCE CURRICULUM
Company	RFP Number	RFP Title
[signature]	CEO	2/26/2021
Signature (required)	Title	Date

(Proceed to the next page only if Confidential Treatment is requested.)

Attachment #4
Response Check List

RFP REFERENCE SECTION	RESPONSE INCLUDED	
	Yes	No
One Electronic Copy	X	
One (1) Electronic Public Copy with Confidential Information Excised (optional)	N/A	
Technical Proposal	X	
Exhibit 1 – Transmittal Letter	X	
Exhibit 2 – Executive Summary	X	
Exhibit 3 – Firm Proposal Terms	X	
Exhibit 4 – Respondent Background information	X	
Exhibit 5 – Experience	X	
Exhibit 6 – Termination, Litigation, and Debarment	X	
Exhibit 7 – Criminal History and Background Investigation	X	
Exhibit 8 – Acceptance of Terms and Conditions	X	
Exhibit 9 – Certification Letter	X	
Exhibit 10 – Authorization to Release Information	X	
Exhibit 11 – Mandatory Technical Specifications	X	
Exhibit 12 – Program Overview	X	
Exhibit 13 – Curriculum	X	
Exhibit 14 – Professional Development	X	
Exhibit 15 – Standards	X	
Exhibit 16 – Evidence of Effectiveness	X	
Exhibit 17 – Optional Features	X	
Exhibit 18 – Addendums	X	
Exhibit 19 – Request for Confidentiality	X	
Cost Proposal	X	