Unit 1 - Problem Solving

The Problem Solving unit is a highly interactive and collaborative introduction to the field of computer science, as framed within the broader pursuit of solving problems. Through a series of puzzles, challenges, and real world scenarios, students are introduced to a problem solving process that they will return to repeatedly throughout the course. Students then learn how computers input, output, store, and process information to help humans solve problems. The unit concludes with students designing an application that helps solve a problem of their choosing.

Chapter 1: The Problem Solving Process

Big Questions

- What strategies and processes can I use to become a more effective problem solver?

Week 1

Lesson 1: Intro to Problem Solving

Unplugged

The class works in groups to design aluminum foil boats that will support as many pennies as possible. At the end of the lesson groups reflect on their experiences with the activity and make connections to the types of problem solving they will be doing for the rest of the course.

Lesson 2: The Problem Solving Process

Unplugged

This lesson introduces the formal problem solving process that the class will use over the course of the year, Define - Prepare - Try - Reflect. The class relates these steps to the aluminum boats problem from the previous lesson, then a problem they are good at solving, then a problem they want to improve at solving. At the end of the lesson the class collects a list of generally useful strategies for each step of the process to put on posters that will be used throughout the unit and year.

Lesson 3: Exploring Problem Solving

Unplugged

In this lesson the class applies the problem solving process to three different problems: a word search, a seating arrangement for a birthday party, and planning a trip. The problems grow increasingly complex and poorly defined to highlight how the problem solving process is particularly helpful when tackling these types of problems.

Chapter Commentary

This chapter guides students to develop and adopt a more formal structured problem solving process by reflecting on problems they have problems they have encountered, both in the classroom and everyday life. By working through a diverse set of problems, such as logic puzzles, engineering challenges, and planning a trip, students learn to identify different classes of problems, decompose large problems, and develop their personal problem solving skills.

Chapter 2: Computers and Problem Solving

Big Questions

- How do computers help people to solve problems?
- How do people and computers approach problems differently?
- What does a computer need from people in order to solve problems effectively?

Week 2

Lesson 4: What is a Computer?

Unplugged

In this lesson the class develops a preliminary definition of a computer. After brainstorming the possible definitions for a computer, the class works in groups to sort pictures into "is a computer" or "is not a computer" on poster paper and explain their motivations for choosing some of the most difficult categorizations. The teacher then introduces a definition of the computer and allows groups to revise their posters according to the new definition.

Lesson 5: Input and Output

In this class students consider a number of computing devices to determine what types of inputs and outputs they use. Groups are assigned to a computing device and based on a teacher-provided definition of input and output, list the inputs and outputs of their device. To conclude the lesson the class examines common activities they do on a computing device and select the inputs and outputs used for that activity from the chart.

Lesson 6: Processing

Unplugged

This lesson dives deeper into the concept of processing that was introduced as part of the definition of a computer. Pairs work together to put a deck of cards in order, a form of processing information. In the end, the class discusses what processing means within the context of solving information problems.

Week 3

Lesson 7: Apps and Storage

Unplugged

This lesson covers the input and output aspects of computers in a context that is relevant and familiar to students: apps. The class evaluates various web applications to analyze the specific problems that they were designed to solve, the inputs that they need to work, and the outputs they provide to users. The class concludes with observations of these apps as well as a teacher led discussion about the impact of apps on society.

Lesson 8: Project - Propose an App

Unplugged | Project

To conclude the study of the problem solving process and the input/output/store/process model of a computer, the class proposes apps designed to solve real world problems. This project is completed across multiple days and culminates in a poster presentation highlighting the features of each app. The project is designed to be completed in pairs though it can be completed individually.

Chapter Commentary

In the second half of the unit, students move on to thinking about computers as machines that solve information problems. Students begin by building a common definition for a computer that focuses on functionality instead of specific hardware. They then explore the ways that computers approach problems. For their final project, students propose an app that could be used to solve a problem of their choosing.

If you are interested in licensing Code.org materials for commercial purposes,**contact us.**

Lesson 1: Intro to Problem Solving

Unplugged

Overview

In this lesson, students work in groups to design aluminum foil boats that will support as many pennies as possible. Groups have two rounds to work on their boats, with the goal of trying to hold more pennies than they did in round 1. The structure of the activity foreshadows different steps of the problem solving process that students will be introduced to in more detail in the following lesson. At the end of the lesson students reflect on their experiences with the activity and make connections to the types of problem solving they will be doing for the rest of the course.

Purpose

This lesson is a fun introduction to the open-ended, collaborative, and creative problem solving students will be using over the rest of this unit and course. The aluminum boats problem could easily be substituted out for any number of other problems that require students to define their goals, devise a plan, try a solution, evaluate their results, and then iteratively improve from there. The fact that the problem chosen is "non-computational" is intentional. Computer science is fundamentally a problem-solving discipline and staying away from traditional computer science problems at this points helps to frame this class as one about problem-solving more generally with computer science being a new "tool" to help attack certain types of problems.

Assessment Opportunities

1. **Communicate and collaborate with classmates in order to solve a problem**

   In the wrap up discussion question about collaboration, check for advantages to collaboration in problem solving and can list appropriate strategies for overcoming challenges. This practice will be reinforced throughout the unit and course.

2. **Iteratively improve a solution to a problem**

   In the Activity Guide, check for strengths and weaknesses in students' original design and connection to appropriate changes for the next iteration of the project.

3. **Identify different strategies used to solve a problem**

   In the Activity Guide, check for strategies for overcoming the challenges in the activity. You also may want to check the "strengths" of their designs that students list as part of developing

Objectives

Students will be able to:

- Communicate and collaborate with classmates in order to solve a problem
- Iteratively improve a solution to a problem
- Identify different strategies used to solve a problem

Preparation

For each group

- ☐ 2 sheets of aluminum foil, 5x5 inches in length each
- ☐ 1 container that can hold 3-5 inches of water
- ☐ Several paper towels or rags that can be placed under the container
- ☐ 15 pennies
- ☐ One copy of the activity guide

For the teacher

- ☐ 1 container that can hold 3-5 inches of water
- ☐ 50 pennies
- ☐ Extra paper towels or rags

Links

> **Heads Up!** Please make a copy of any documents you plan to share with students.

For the Teacher
- **Aluminum Boats** - Exemplar
- **Code.org How-to Video Playlist**

For the Students
- **Aluminum Boats** - Activity Guide
  - Make a Copy ▾

their plan on the second page of the guide.

Agenda

Tech Setup

 Before class
 At the beginning of class

CSD Pre-Course Survey

 Important! Have your students take the CSD Pre-Course Survey!

Warm Up (10 min)

 Set the Stage

Activity (30 min)

 Building an Aluminum Boat
 Iterative Improvement

Wrap Up (10 mins)

 Discuss the Challenge

Teaching Guide

Tech Setup

Throughout the year, in order for you to be able to see student work, and for students to be able to access online tools and resources, all students need to be in the same section on Code Studio. To see assessments and answer keys that Code.org provides, you also need to be a "verified teacher".

Before class

Ensure you are registered on Code Studio as a "verified" teacher account

Anyone can create a teacher account on Code Studio, which means that we need an extra layer of authorization to allow CS Discoveries teachers to see assessments, answer keys, and any other collateral that students should not be able to trivially get access to. If you attended a Code.org TeacherCon during the summer, you should already have this access.

To check whether you have access:

1. Navigate to the **Unit 1 course overview page**
2. Do you see the CS Discoveries Pre-course Survey as the first "stage" of the course?
3. If not, please fill out **this form**. Note that it can take a day or so to become a verified teacher account, so please do this step early!

If you are **not** a verified teacher account, you can still create a section for your class, but you will not be able to administer the pre-course survey on the first day.

Create a class section on Code Studio.

To create a section:

1. Navigate to the **Teacher Home Page**
2. Click 'New section' under 'Classroom Sections'
3. Choose 'Email Logins'
4. In the space provided, give your section a name (e.g. CSD Period 1), the grade of your students, Course: 'Computer Science Discoveries', Current Unit: 'Unit 1: Problem Solving', and click 'Save'
5. Once the section is created, click the name of the section
6. On the new page, you will see a unique Join URL that you will distribute to your students in class.

For a video walkthrough of these steps and more on navigating your Code.org account, go to the **Code.org How-to Video Playlist**

At the beginning of class

1. Have students create a Code Studio account at **https://studio.code.org** if they don't already have one
2. Share the section Join URL with students and tell them to navigate to it to join your section
   - You can confirm that a student successfully joined your section by having the section progress page on the Teacher Home Page open and hitting refresh as students join. Students should see a small green bar at the top of their page that says 'You've successfully joined ...'
3. From **studio.code.org** have students locate the Computer Science Discoveries tile and click 'View course', then go to Unit 1.

Once students are looking at the Unit 1 overview page, they will be ready to take the CSD Pre-Course survey

CSD Pre-Course Survey

Important! Have your students take the CSD Pre-Course Survey!

Students can find a link to the survey in Code Studio as the first item on the Unit 1 overview page. To ensure that students only take the survey at the appropriate time, it is "locked" and unviewable by them until you "unlock" the survey. The **How to Administer a Locked Assessment** document provides details on how to do that when you are ready. Note that the instructions for administering an assessment and a survey are the same.

How much time does it take?

The survey does take some time - it is roughly 30 questions. You might consider administering it on an admin day at school, or as an early homework.

Because it is a pre-course survey it is important that students take it as early in the course as possible, before they have had much (or any) exposure to the class so that we may accurately gauge changes in attitudes and beliefs caused by the course.

Thank you!

-- The Code.org Team

- A major goal of CS Discoveries is to broaden participation in computer science.
- It is **crucial** therefore to have insight into students' attitudes and beliefs about computer science **before** the course so that we can measure the amount of change that occurred **after** the course is over.
- Please note that this survey is **anonymous for students**
- Completing it also helps us understand important improvements we can make to the curriculum to improve the teacher and student experience.

Please help by having your students contribute to this vital dataset. Their voices make the difference!

Warm Up (10 min)

Set the Stage

💡 🎤 *Remarks*

> Hello everyone, welcome to Computer Science Discoveries! You may have a lot of questions about what we're going to do or learn this year. All I'll say for now is that we're going to be having a lot of fun learning how we can use computer science to help us build things, express ourselves, and solve problems. In fact, solving problems with or without computers is going to be one of the primary focuses of this class. But enough of me talking, let's get to today's activity.

Group: Put students in groups of 2 or 3.

💡 **Distribute**: One copy of the activity guide to each group. As a class, read through the Goal and Rules sections of the activity guide and answer questions.

💡 Teaching Tip

Feel free to use any icebreaker activity or other introduction you like here. The goal is to move quickly to the main activity after which you'll have more opportunity to reflect about problem solving in the context of today's activity.

💡 Teaching Tip

Getting Copies of Google Docs: Activity Guides in this curriculum are available as Google Docs, Word Docs, and PDFs. If you'd like to edit or make your own copy of a Google Doc you may without requesting access. Simply log in with a Google account, click "File" and "Make a copy" from the menu. You may then share your own copy with your students.

Activity (30 min)

Building an Aluminum Boat

Prompt: Today we're going to be building aluminum boats. You'll have an opportunity to build at least two boats and use your experience with each one to improve your designs. Before we get started, decide as a group what kind of design you'd like to make with your first boat. Record your ideas and any possible weaknesses of this design on your activity guide.
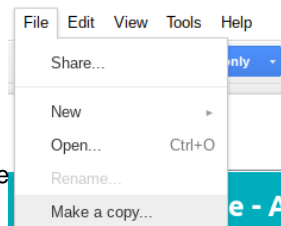
💡 **Support:** Give students a couple minutes to discuss in groups the approach they will be taking with this first boat. Once groups have recorded their ideas and some possible weaknesses they can come to you to get their aluminum foil and begin building their boats.

Once groups are ready, have them test their boats by dropping individual pennies into the boat. Remind them of the rules, specifically that they can't touch or adjust the boats once they're in the water. Have them record the total number of pennies held on their activity guides.

Iterative Improvement

🎤 *Remarks*

> This first attempt at building our boats was just to get familiar with the challenge. We're all going to build a second boat and see if we can improve the number of pennies our boats held. Before we get started though, let's see what we can learn from this trial run.

Share: Have students share the results of their first run with neighboring groups. Ask groups to focus particularly on what the eventual failure of their boat was (e.g. it wasn't deep enough, it was unstable, etc.) and brainstorm ways to get around those problems.

Prompt: Now that you've had a chance to learn from the first round of boatmaking, let's run the same activity again. First, your group will develop a new plan. Just as before, record it on your activity guide, and once you're ready I'll come around and give you a new piece of foil.

Support: As you circulate from group to group, ask questions about the group's focus in redesign. EX: "What aspect of your boat needed the most improvement?" "What ideas from other groups did you want to incorporate to yours?" "Did you feel the need to completely restructure your boat, or make minor modifications?"

Once groups have prepared their new plans give them a new piece of foil and have them each build a new boat.

Groups can test their designs just as before and record the results on their activity guides.

Transition: Ask class to return to their own seats to reflect on the activity.

> 💡 **Teaching Tip**
>
> This activity can get your room wet! Ideally, each group should have its own bucket/container with water to conduct its own tests. If needed, you can have several groups share one container, but be mindful to set guidelines for sharing that container. Place towel/rag under each container. You may also want to consider to have this activity in the hallway or some other space if your room is very restrictive.

> 💡 **Teaching Tip**
>
> Hold onto the foil until students submit a plan for their boat. The goal isn't to slow them down too much, but just give them a moment to reflect briefly on the possible approaches they could take. This is one way this activity foreshadows the Planning step of the problem solving process students will see in subsequent lessons.

> 💡 **Teaching Tip**
>
> While some students will view this portion as a competition, emphasize that each group is looking to improve its own design, not competing against others. You are appealing for each student to challenge themselves first, not others.

Wrap Up (10 mins)

Discuss the Challenge

Allow students to discuss one or more of the following questions. Consider giving students time to reflect individually in a journal or an exit ticket before having them share their thoughts with each other.

Prompt: What was your favorite part of this activity? What was most challenging?

Discuss: Allow students time to share thoughts in small groups before sharing out ideas as a class.

Prompt: You worked in groups for this activity. How did working in a group make this activity easier, how did it make the activity more challenging? What helped your group overcome these challenges?

Discuss: Allow students time to share thoughts with the class.

> 💬 **Discussion**
>
> This should be a fairly open-ended discussion of the different components of the activity. Feel free to ask follow up questions if you like but the main goal is just to kick off the later conversation.

- **Prompt**: Since you are in a computer science class, you also may be wondering, "What in the world did that have to do with computer science?" Find another partner and talk about what you think this activity has to do with computer science.

- **Discuss**: Allow students an opportunity to share their responses with the class.

🎤 *Remarks*

> All of your thoughts around these questions were great. You may be used to thinking about computer science as being all about computers. I'm here to tell you that first and foremost computer science is about solving problems, and that's what we were doing today. A lot of other parts of this activity like improving designs, working in groups, and building things is also going to be a big part of this class. I hope you're excited for the year. Tomorrow we'll start digging deeper into problem solving itself.

✔ Assessment Opportunity

Students may recognize that collaboration allowed them to think through their ideas more thouroughly or that they were able to split up the work that they needed to do. As students list challenges, ensure that they are coming up with ways to overcome those challenges. If you have noticed any issues in group dynamics during the activity, you may want to use this time to bring up these as potential problems in a non-confrontational way and help students to generate strategies for working together more effectively in future activities. These can serve as classroom norms for group work moving forward.

💬 Discussion

The second question will give you a chance to hear some of the thoughts and beliefs students have about computer science. Students may suggest things like, teamwork, sharing ideas, making improvements, etc. Again, the goal is not to agree or disagree, but to foster an environment to share those thoughts. It is appropriate to ask follow up questions that are not threatening, such as, "explain that thought to me more".

🎓 Content Corner

Although there are no right or wrong answers for this discussion, for our purposes, the main point is that students solved a problem. They had to define the problem, plan a solution, try a solution, and evaluate it. These concepts will become more apparent as the chapter unfolds.

Standards Alignment

CSTA K-12 Computer Science Standards (2017)

▶ **AP** - Algorithms & Programming

Lesson 2: The Problem Solving Process

Unplugged

Overview

This lesson introduces the formal problem solving process that students will use over the course of the year, Define - Prepare - Try - Reflect. The lesson begins by asking students to brainstorm all the different types of problems that they encounter in everyday life. Students are then shown the four steps of the problem solving process and work together to relate these abstract steps to their actual experiences solving problems. First students relate these steps to the aluminum boats problem from the previous lesson, then a problem they are good at solving, then a problem they want to improve at solving. At the end of the lesson the class collects a list of generally useful strategies for each step of the process to put on posters that will be used throughout the unit and year.

Purpose

This lesson aims to anchor the formal problem solving process students will use throughout the course in some real-life experiences they already have solving problems. Future units in CS Discoveries will present problems in contexts that may or may not be familiar. A structured problem solving process will be an important tool for helping students move forward in the face of novel and complex challenges.

Assessment Opportunities

1. **Given a problem, identify individual actions that would fall within each step of the problem solving process**

   On page 2 of the activity guide, check that students have written down reasonable actions for each of the four steps in the problem they are trying to solve.

2. **Identify useful strategies within each step of the problem solving process**

   In the wrap up, check that the groups are coming up with effective strategies for each step of the process. You may also want to collect the strategies that each group wrote down as an exit ticket.

Agenda

Warm Up (5 min)

   Problems Brainstorm

Activity (30 min)

   Introduce the Problem Solving Process

View on Code Studio

Objectives

Students will be able to:

- Given a problem, identify individual actions that would fall within each step of the problem solving process
- Identify useful strategies within each step of the problem solving process

Preparation

For each student

   ☐ Print a copy of Activity Guide

For the class

   ☐ Poster paper
   ☐ Markers/colored pencils

Links

> **Heads Up!** Please make a copy of any documents you plan to share with students.

For the Teacher

- **Problem Solving Process** - Exemplar
- **Problem Solving Process** - Graphic

For the Students

- **Problem Solving Process** - Video (**download**)
- **The Problem Solving Process** - Activity Guide Make a Copy ▾

The Problem Solving Process in Context

Wrap Up (15 min)

Create Posters of the Steps

Extended Learning

Article Discussion

Teaching Guide

Warm Up (5 min)

💬 Problems Brainstorm

Prompt: We use the term "problem" to refer to lots of different situations. I could say I have a problem for homework, a problem with my brother, and a problem with my car, and all three mean very different things. On a sheet of paper I want you to brainstorm as many different kinds of problems as you can and be ready to share with the class.

💡 **Discuss:** Students should silently record their ideas in writing for a couple minutes. Afterwards invite them to share what they wrote with a neighbor and then finally bring the whole class together to develop a classwide list. Record all the different kinds of problems students think of on the board or somewhere else that they'll be clearly visible.

🎤 *Remarks*

> Clearly we encounter problems in lots of different areas of our lives. Depending on the context, this word can have many different meanings. For now let's just say that a problem is a situation that could be fixed or improved.

Activity (30 min)

Introduce the Problem Solving Process

🎤 *Remarks*

> We solve problems all the time, but we don't often think about how we're solving problems. Having a strategy or process to approach lots of different kinds of problems can make you a more thoughtful, creative, and successful problem solver.

Distribute: Hand out copies of the activity guides.

The Problem Solving Process in Context

Video: Watch the Problem Solving Process Video on Code Studio

🖥 Code Studio levels

- Video: Problem Solving Process 🎥
- **Teacher Overview**
- **Student Overview**

Discussion Goals

View on Code Studio
🔗

The first discussion question is part of the core activity in the rest of the lesson. Students should take time working int heir groups to think of specific things that they did that follow each step of the process. The goal of this discussion is to make the abstract steps more concrete and accessible to students by relating them to a shared activity.

The second question gives students a chance to expand their understanding of the process to a different problem, seeing how the different steps my look in different domains. The goal of this discussion is to makes sure students have a general enough understanding of the process that they can apply it to a wider variety of problems, since the process will be used throughout the course in various domains.

Questions to consider

- How did you follow the problem solving process in the last lesson?
- How could you use this process on a problem in your everyday life?

Step 1: Introduce and as a class review the descriptions of the four steps in the process by reading them aloud. Answer or discuss any questions students have about the process but otherwise move on to completing the first section of the activity guide.

Step 2: Have students complete the first section of the activity guide by filling in the steps of the previous day's activity they think fall within each step of the problem solving process.

💬 **Discuss:** Once students have completed the first section of the activity guide ask them to share with neighbors and then with the class as a whole.

Step 3: Ask students to select one type of problem that they think they're really good at solving. Use the list of problems already on the board to help students think of their type of problem. Again give them a couple of minutes to quietly record the steps of their process before sharing with a neighbor.

💬 **Discuss:** Have students share what they wrote with a neighbor and then once again lead a discussion of the conversations they had. Ask students to talk about the individual steps they're using to solve their chosen problem but also point out instances where the same types of strategies are appearing multiple times.

Step 4: Place students in pairs and ask them to complete the final section of the activity guide. They will need to choose a type of problem that both members of the group want to get better at solving and then write the steps they would use within the problem solving process to solve that problem.

Discuss: Lead one final share out in which students present how they would use the problem solving process to approach a less familiar problem.

💬 Discussion

Goal: For this first conversation in particular you're making sure students understand the meaning of the four different steps. While some steps might sometimes be categorized in two ways, use this chance to talk about that ambiguity. Your goal is to use the shared context of the aluminum boats problem to understand this process. Here's a possible set of steps students may come up with.

Define: Understanding the problem when it was assigned, examining available resources, finding problems with their original design before deciding how to fix them, looking at problems with other groups' boats

Prepare: Discussing with team members how to proceed, brainstorming approaches, anticipating possible flaws.

Try: Actually building the boats, running the test

Reflect: Examining the results of their test, comparing their results to their predictions, discussing with group members the reasons the boat sunk eventually.

💬 Discussion

Goal: All three of these discussions aim to reinforce the meaning of the 4 steps in the problem solving process. In this discussion you might lean more heavily on other students to ensure that the strategies and steps being offered by students seem to fit the definitions of the 4 steps provided on the activity guide.

Wrap Up (15 min)

Create Posters of the Steps

Set-up: At the front of the room place four large posters with the names of one of the steps of the problem solving process written on each.

Prompt: At your tables review all the work you did today looking at the problem solving process in a number of contexts and pick the two most important strategies for each step in the process. These should be strategies that you think can help in lots of different types of problems when you're working on that step.

⊘ **Circulate:** Walk around the room and check that groups are making progress on picking their strategies. Remind them that these are supposed to be generally useful and not specific to a single type of problem. Once all groups are ready bring the class back together

⊘ Assessment Opportunity

Check that the groups are coming up with effective strategies for each step of the process. You may also want to collect the strategies that each group wrote down as an exit ticket.

Share: Go through each step of the problem solving process and ask groups to share their strategies. At the front of the room record the strategies on the appropriate poster. Once all posters have been completed place them somewhere visible in the room.

🎤 *Remarks*

> I began by saying a formal problem solving process could help us solve all kinds of problems. Today we began to understand what this process looks like in a variety of real life situations. Tomorrow we're going to start putting this process into action to see how it actually works.

Extended Learning

Article Discussion

Read through the article, **You Are Solving the Wrong Problem**

1. What was interesting about this article?
2. What current events do you think we need to look at through this problem solving process? Why?

Standards Alignment

CSTA K-12 Computer Science Standards (2017)

▶ **AP** - Algorithms & Programming

Lesson 3: Exploring Problem Solving

Unplugged

Overview

In this lesson students apply the problem solving process to three different problems in order to better understand the value of each step. They will solve a word search, arrange seating for a birthday party, and plan a trip. The problems grow increasingly complex and poorly defined to highlight how the problem solving process is particularly helpful when tackling these types of problems. The lesson concludes with students reflecting on their experience with the problem solving process. They will justify the inclusion of each step and will brainstorm questions or strategies that can help them better define open-ended problems, as this is often the most critical step.

This lesson will likely take two class periods or more to complete. The first two problems may fit into a single class period but the third will need to be moved to a second day.

Purpose

This lesson provides students with more practice with the problem solving process in a variety of contexts. It highlights the fact that the problem solving process is particularly helpful when approaching poorly defined problems. The final brainstorm of the lesson provides students with some strategies and questions they can ask to better define problems for themselves, since this is often the most critical step. The problems seen in this lesson also help to drive a discussion in the following lesson about the types of problems that computers are well-suited to solve.

Assessment Opportunities

1. **Apply the problem solving process to approach a variety of problems**

   On page 5 of the Activity Guide, check the chart to make sure that students are putting in reasonable steps for each of the problems.

2. **Assess how well-defined a problem is and use strategies to define the problem more precisely**

   For the last question on page 5 of the Activity Guide, make sure that students have at least one question or strategy that would be an effective way to better define a problem.

Agenda

Warm Up (5 min)

    Setting the Stage

View on Code Studio

Objectives

Students will be able to:

- Apply the problem solving process to approach a variety of problems
- Assess how well-defined a problem is and use strategies to define the problem more precisely

Preparation

☐ Print the activity guide for each student
☐ Spend a few minutes exploring **Google Maps - Website**
☐ Scratch paper for the Birthday Party problem
☐ Poster to record strategies for defining problems in wrap up discussion

Links

> **Heads Up!** Please make a copy of any documents you plan to share with students.

For the Teacher

- **Solving Problems** - Exemplar
- **Google Maps** - Website

For the Students

- **Solving Problems** - Activity Guide
  Make a Copy ▾

Activity (75 min)

    Solving Problems

Wrap Up (20 min)

    Understanding the Problem Solving Process

Teaching Guide

Warm Up (5 min)

Setting the Stage

♀ 🎤 *Remarks*

> Yesterday we talked about many different types of real life problems and learned the four steps of the problem solving process. With such a wide variety of problems and strategies, it's important to be able to think critically about how best to use the problem solving process. Today we're going to look at a wide variety of problems, talk about what makes them different, and reflect on how the problem solving process helped us solve them.

Group: For all three activities students should be working together in groups, even if they record their results individually. Groups of 2-4 will likely work best.

Distribute: Hand out the activity guide, one copy for each student. For now they can be face down so that the word search isn't visible.

Activity (75 min)

Solving Problems

♀ ✎ *Word Search*

> Once students are in pairs ask them to flip over their activity guides and begin the first challenge. They'll be finding the 8 words in a 20 by 20 grid of letters.
>
> **Circulate:** Walk around the room observing how students are addressing the problem. Make sure that groups are not sharing locations of words. Encourage them to think about how making a plan might help them address this task.
>
> Once all groups have finished, bring the class back together. Have students flip to the last page of the activity guide where there is a table to record their experiences with the problem. They will record what parts of solving this problem fall within each step of the problem solving process.

Discuss: Briefly discuss with students what parts of the activity they felt fell into each step of the problem solving process. Some possible points to make after students share are below.

- **Define**: This problem was already very well defined. Not all problems will be, though.
- **Prepare**: Developing a plan with a team (such as divvying up the words, splitting the grid into separate sections that each member searches in, or just being methodical about looking for words) makes this problem much easier to solve than random searching.
- **Try**: Patience and persistence is important to see your plan through
- **Reflect**: If your early plans are not working you can regroup and choose a new plan

♀ ✎ *Birthday Guests*

Move the class on to the birthday guests problem. Groups may still work together on their solutions but shouldn't share with other groups.

Circulate: As before, circulate around the room noting the types of strategies that groups are using. Remind them to use the steps of the problem solving process to help them if they're getting stuck.

Once groups have finished solving the problem ask them to move to the last page of the activity guide to record how they used the problem solving process to solve this problem.

Discuss: Briefly discuss with students what parts of the activity they felt fell into each step of the problem solving process. Some possible points to make after students share are below.

- **Define**: The problem seems to be a problem of seating individuals. If you instead think of it as a problem of seating groups of people who would like to be together there are many fewer possible solutions to consider.
- **Prepare**: Ask students to share what types of strategies they considered before just starting to assign people to seats.
- **Try**: As before, patience and persistence is important to see your plan through
- **Reflect**: If early strategies are not working groups may have regrouped and tried a more structured approach

💡 📝 *Plan a Trip*

🖥 Students will need to work online for this problem to use **Google Maps - Website** or some other tool that will allow them to plan a trip.

Demonstrate: Move the class on to the Plan a Trip problem. Each member of the group will individually be developing a plan for a trip that follows criteria they'll develop as a team. Before sending groups off you'll want to demonstrate how the tool they're using will work. A good set of steps to show them might be.

- Find your school on the map.
- Search for something familiar to students close to the school.
- Generate directions from the school to the other location, choosing the method of travel (walking/bus/car/etc.)
- Highlight where the tool shows the total time of the trip

Prompt: Give students time to choose the goals they'll use to plan their trip. For example there may be certain kinds of activities they'd like to do, places they definitely want to include, people they want to visit, etc.

Circulate: Once groups have goals, they will move through the activity by developing a plan to visit the places in one school day. Give them a time limit on this part of this process, e.g. 15 minutes, to make sure they focus on the key elements on their plan rather than perfecting it. They should record key information about their plan in their activity guides.

Bring groups back together and have them share their initial plans. On the activity guides they can record the feedback their classmates give them on their plans.

Once groups have discussed what they like or don't like about their classmates' proposed plans, they can re-examine them and make improvements. Are there other things they'd like to do? Do they have new goals? Give them several minutes to make improvements to their plans before deciding on a final version.

Bring the class back together and have them record the different steps of the problem solving process that they used in their activity guides.

Discuss: Briefly discuss with students what parts of the activity they felt fell into each step of the problem solving process. Some possible points to make after students share are below.

- **Define**: This problem was not well-defined. They needed to decide for themselves what a "good" itinerary looked like, and this definition could have even shifted throughout the process.
- **Prepare**: Narrowing down a list of possible destinations is helpful. You may also choose to make the point that this entire activity is an example of preparation. You can't go on every possible trip and then pick the best one, so you need to do the kind of planning they're doing here.
- **Try**: As before, patience and persistence is important to see your plan through
- **Reflect**: In this problem reflection came primarily through feedback from peers. Some destinations might not end up being that interesting to other group members. Some trips are fun but require too much driving. Feedback is an important part of the reflect step, especially in group work.

Wrap Up (20 min)

💬 Understanding the Problem Solving Process

Prompt: You just solved a number of very different problems. With your tables review the notes you took on each of the problems. Be ready to report out on the following questions

- For each step in the problem solving process, what is their purpose? Why are they included?
- Are there any kinds of problems that the problem solving process is particularly helpful at solving?

Discuss: After tables have discussed their responses for several minutes invite the whole class to share their rationale for including each step in the process. Once each step has been discussed, move on to the second question. This question may have many responses and you should allow students to share their thoughts and experiences. If it doesn't arise naturally as you leave the conversation offer some or all of the ideas mentioned in the discussion goals.

💬 **Prompt:** The problem solving process is particularly helpful when we encounter poorly-defined problems. We saw today that without a well-defined problem the rest of the problem solving process is difficult to follow. What are some questions or strategies we can use to help us better understand and define problems before we try to solve them.

💬 Discussion

Goal: Students have practiced using the problem solving process on a number of different problems. Help them synthesize the notes they have been keeping to better understand the role of each step and the value of the problem solving process in general. A sample set of conclusions is below but you should allow students to share your own insights before offering your own.

Define: without defining a problem you might solve the wrong problem, not know where to start, or not know when you're finished

Prepare: Even well-defined problems usually have many possible approaches. Make each try more likely to succeed by first examining your options and anticipating challenges

Try: Without trying you'll never get anywhere. It's important to be persistent and patient so long as your plan still may work

Reflect: You'll likely not solve the problem the first time or there will be a better way to solve it. Learn from your past attempts and get ready to start the process again.

The Problem Solving Process: While you may notice you're using it even for small and trivial problems, this process is incredibly useful for large, complex, poorly-defined, or open-ended problems. It helps you make progress when the way forward may not always be clear.

Discuss: Have groups share quickly before taking suggestions from the class as a whole and recording them on a poster. Ensure that the three strategies indicated in the discussion goal also make their way onto the poster.

🎤 *Remarks*

Excellent work everyone. We now understand a great deal about the problem solving process. This is an incredibly useful tool that we'll use repeatedly throughout the year as we dig deeper into understanding the world of computer science.

💬 Discussion

Goal: As a final closing to the lesson, highlight the fact that defining a problem well often makes the rest of the process much easier. This brainstorm should result in a poster of questions or other kind of shared list that you can point to throughout the year to help students better define problems.

Connections: On the final project of the unit students will be asked to use some of these questions to better define a problem of their choosing. In particular they will be asked to consider:

- Who in particular the problem affects. What specifically do they need? In what kind of situations?
- Why the problem exists? (And why does that problem exist?) Keep asking to get to the heart of the problem.
- How could I be able to tell the problem had been solved? What could I observe or measure?

You can add these questions to the poster at the end of the conversation if they do not naturally arise.

Standards Alignment

CSTA K-12 Computer Science Standards (2017)

▶ **AP** - Algorithms & Programming

If you are interested in licensing Code.org materials for commercial purposes,**contact us.**

Lesson 4: What is a Computer?

Unplugged

Overview

In this lesson students develop a preliminary definition of a computer. To begin the lesson, the class will brainstorm possible definitions for a computer and place the results of this brainstorm on the board. Next, students will work in groups to sort pictures into "is a computer" or "is not a computer" on poster paper. Groups will place their posters around the room and briefly explain their motivations for choosing some of their most difficult categorizations. The teacher will then introduce a definition of the computer and allow students to revise their posters according to the new definition.

Purpose

This lesson serves as a transition out of the first chapter of Unit 1 and into the second chapter. Up until now, students have considered the potential of computers as problem solving devices, but not established what a computer actually is. In this lesson, students will consider different types of computers and that these computers input, store, process, and output information as part of the problem solving process. Upcoming lessons will dive much deeper into what an information problem looks like and how computers solve these problems.

Assessment Opportunities

1. **Identify a computer as a machine that processes information**

   In the last discussion of the main activity, check students' reasoning about whether a device is a computer. Make sure that they are mentioning that it processes information or solves information (or thinking) problems.

2. **Provide a high level description of the different parts of the Input - Output - Store - Process model of a computer**

   In the wrap-up journal prompt, check that each student's descriptions of the different parts of the IOSP model accurately describe the input, output, storage, and processing in regards to the problem. Don't worry if they are not specific in the processing aspect. Later lessons in the unit will go over processing in more detail.

Agenda

Warm Up (5 mins)

View on Code Studio

Objectives

Students will be able to:

- Identify a computer as a machine that processes information
- Provide a high level description of the different parts of the Input - Output - Store - Process model of a computer

Preparation

For each group

☐ Print out copies of the activity guide. Note there are two sets of pictures, but each group only needs a single set.
☐ Scissors (if you will not have time to cut the pictures prior to class)
☐ Poster paper
☐ Markers or colored pencils
☐ Glue or tape to attach pictures

Links

> **Heads Up!** Please make a copy of any documents you plan to share with students.

For the Teacher

- **What is a Computer** - Exemplar
- **What Makes a Computer, a Computer?** - Video
- **What is a Computer?** - Graphic

For the Students

- **What is a Computer (Version B)** - Activity Guide [Make a Copy ▾]
- **What is a Computer (Version A)** - Activity Guide [Make a Copy ▾]

What Problems Do Computers Help You Solve?

Activity (40 mins)

Computer or Not?
Present Your Categorizations

Wrap Up (5 mins)

Journal

Teaching Guide

Warm Up (5 mins)

What Problems Do Computers Help You Solve?

Prompt: In the modern day we use computers almost constantly. What kinds of problems do computers help you solve? How do they help you do this?

💬 **Discuss:** Run this conversation as a brainstorm, recording ideas on the board. Note and call out similarities you're seeing in the kinds of problems students identified.

🎙️ *Remarks*

> Computers are clearly an important part of our lives and help us solve all kinds of problems. I want to think more about the kinds of problems computers help us solve, but first I want to ask an important question. What is a computer?

💬 Discussion

Goal: This warm up makes the transition from thinking about problem solving in a generic sense to thinking about how computers help solve certain kinds of problems. While the lesson will eventually reveal that computers are particularly useful at solving information problems, you don't need to make that point during this brainstorm.

Activity (40 mins)

Computer or Not?

Group: Place students in groups of 3 or 4

💡 **Distribute:** Activity Guide as well as scissors, markers / colored pencils, poster paper, and glue / tape for making posters. (Note that there are two possible versions of the activity guide. Choose the best for your class, or give different groups different versions.)

Give students the following directions:

- Draw a line down the middle of your poster, label one side "Computer" and the other "Not a Computer"
- Discuss as a group which of the objects in your set (from the activity guide) belong in each category
- Once your group is in agreement tape your objects to the appropriate side
- Develop a list of characteristics your groups used to determine whether an object is a computer

💡 Teaching Tip

Modifications from the Forum: Many teachers have shared ideas for modifying this lesson on the forum **(link)**. Head there to check out ways teachers have reduced printables, integrated technology, or otherwise adapted this activity to fit the needs of their class. If you do something new, share your ideas too!

💡 **Circulate:** Circle the room as students work to categorize the different images on the activity guide. Encourage groups to talk openly about their ideas and explain why they do or don't think an object should be categorized as a computer. For groups that can't decide on a categorization, ask members to defend their points of view, and take a majority vote. Assure groups that it is ok if one or two people disagree.

💡 Teaching Tip

Tape First: Students will have an opportunity to update their categorizations later in the lesson. For now they should just tape their objects to their poster or even just place them on the correct side.

At the end of the time bring the class back together and ask them to place their posters at the front of the room.

Present Your Categorizations

💡 **Share:** Have each group briefly present their posters, focusing their discussion on the following points

1. What rules or definition did you use to categorize your objects?

2. Which item was most difficult for you to categorize? How did you eventually make the decision of where to place it?

Invite the audience to respectfully question any categorizations if they disagree with the presenting group's decisions.

🎤 *Remarks*

> As you can see, it's not always clear whether something is a computer, and even experts sometimes have different points of view. Let's have a look, however, at a definition that we'll use throughout this

Display: Show **What Makes a Computer, a Computer? - Video**. This video is also available to students on the Code.org website, including an alternative link for schools where YouTube is blocked. The video presents a computer as a machine that helps with certain kinds of thinking work by processing information. It formally introduces a model of a computer as a machine that inputs, outputs, stores, and processes information.

Allow students to revise their posters using the definition they have just learned.

✔ **Discuss:** Did any groups change their minds about whether something was a computer? What about the definition convinced you?

Wrap Up (5 mins)

Journal

💡 **Prompt:** Today you've had a chance to look at a definition of a computer that focuses on how the computer solves problems. We've also seen many different types of computers. In your journal, think of a problem that a computer can help you to solve.

✔ • What is the problem?
   • What information is input to the computer?
   • What information does the computer store?
   • What information does the computer process?
   • What information does the computer output?

Standards Alignment

CSTA K-12 Computer Science Standards (2017)

   ▶ **CS** - Computing Systems

If you are interested in licensing Code.org materials for commercial purposes,**contact us.**

Lesson 5: Input and Output

Overview

In this lesson students consider a number of computing devices to determine what types of inputs and outputs they use. Groups are assigned to a computing device and based on a teacher-provided definition of input and output, list the inputs and outputs of their device. Earlier in the activity students are prompted to focus on more obvious physical inputs and outputs (e.g. a keyboard as an input or a screen as an output) but later discussions lead students to consider less obvious examples (e.g. that a touch screen is both an input and output, or the fact that the Internet can serve as both input and output). Throughout the lesson the teacher records inputs and outputs that are identified on a T-Chart at the front of the room. To conclude the lesson students examine common activities they do on a computing device and select the inputs and outputs used for that activity from the chart.

Purpose

In the previous lesson students were introduced to the input, output, store, and process model of a computer. Since this model of a computer is still fairly abstract, this lesson grounds it in the ways a computer actually input and output information.

Assessment Opportunities

1. **Identify the inputs and outputs of common computing devices**

    On the activity guide, check that each has appropriate inputs and outputs listed. Also, look to see that different devices have different answers listed, such at at least some answers are specific to the particular device in that row.

2. **Select the inputs and outputs used to perform common computing tasks**

    In the wrap up discussion, make sure that students come up with multiple examples of tasks that involve input and output, and that they are coming up with multiple types of inputs and outputs specific to the task at hand. You might also collect individual answers to this question as an exit ticket.

Agenda

Warm Up (5 mins)
    Computer "Stuff" is Information
Activity (30 mins)

View on Code Studio

Objectives

Students will be able to:
*   Identify the inputs and outputs of common computing devices
*   Select the inputs and outputs used to perform common computing tasks

Preparation

☐ Prepare copies of the activity guide

Links

> **Heads Up!** Please make a copy of any documents you plan to share with students.

For the Teacher
*   **Input and Output** - Exemplar

For the Students
*   **Input and Output - Version D** - Activity Guide | Make a Copy ▾
*   **Input and Output - Version C** - Activity Guide | Make a Copy ▾
*   **Input and Output - Version B** - Activity Guide | Make a Copy ▾
*   **Input and Output - Version A** - Activity Guide | Make a Copy ▾

Vocabulary

*   **Input** - A device or component that allows information to be given to a computer
*   **Output** - Any device or component that receives information from a computer

Brainstorming Inputs and Outputs

Wrap Up (15 mins)

What Inputs and Outputs Do I Use?

Teaching Guide

Warm Up (5 mins)

Computer "Stuff" is Information

Prompt: In the last lesson we learned that a computer is a device that "processes information". That phrase is still a little vague, so let's see if we can clarify it a bit. What kinds of "stuff" do you use, look at, create, or edit on a computer that you think might actually be information?

💬 **Discuss:** Students should brainstorm silently, then share in pairs, then discuss their ideas with the whole class. This conversation should be pretty open-ended.

🎤 *Remarks*

> Great work, and I'm sure we could think of more examples if we tried. All the examples you just provided are the "information" that we use computers to process. Today we're going to look at an important question of how this information gets in and out of the computer in the first place.

Activity (30 mins)

Brainstorming Inputs and Outputs

Group: Place students in small groups of 2 or 3.

💡 **Distribute:** Copies of the activity guide to each group. Four versions of the activity guide are available, each with a different device in the final row of the chart. You may choose to give every group the same version of the activity guide, or give different versions to different groups.

Display: At the front of the room place a large poster or project a document where you will record all the inputs and outputs students brainstorm in this activity.

📝 *Input and Output Activity Guide*

Vocabulary: As a class review the two vocabulary words found on the top of the activity guide, input, and output.

📢 **Inputs and Outputs - Desktop:** Have students complete the first row of the activity guide by listing all of the possible inputs and outputs to a desktop. Encourage students to think about the actual computer, as well as devices that they connect to one.

Discuss: Have students share the results of their brainstorm. As they share them record their answers on the poster at the front of the room.

📢 **Inputs and Outputs - Tablet:** Have students complete the second row of the activity guide by listing all of the possible inputs and outputs to a laptop.

Discuss: Have students share the results of their brainstorm. As they share them record their answers on the poster at the front of the room. In particular call attention to the fact that a screen can serve many roles as both the input and output of a tablet. Some less obvious inputs may require you to provide examples to students.

For example, if a tablet shows you where you are on a map then it must somehow know where you are. In other words there must be some kind of input (a GPS system) that provides that information.

Similarly if a tablet knows when it's been rotated then there must be some kind of input device that is detecting how a tablet is positioned.

💡 **Inputs and Outputs - Third device, (e.g. Smartphone):** Have students complete the final row of the activity guide by listing all of the inputs and outputs for the device on their respective activity guides.

Discuss: Have students share the results of their brainstorm. As they share them record their answers on the poster at the front of the room. In particular call attention to the fact different devices need different inputs and outputs to better serve their particular purposes.

Wrap Up (15 mins)

What Inputs and Outputs Do I Use?

Prompt: Brainstorm three everyday activities you or people you know do with a computer.

What is the input used for that activity?

What is the output?

Circulate: Have students brainstorm with their groups and record their ideas on their activity guides. Point students towards the list of inputs and outputs you've listed on the board.

✔ **Discuss:** As a class discuss the examples students brainstorm.

Standards Alignment

CSTA K-12 Computer Science Standards (2017)

▶ **CS** - Computing Systems

🎓 Content Corner

Possible Responses Tablet: The following is a list of possible responses

- **Inputs:** Touch screen, Buttons, Microphone, GPS, Motion sensor (e.g. to rotate the screen), Light sensor (e.g. to make screen dimmer at night), Camera, Internet connection
- **Outputs:** Touch screen, Speakers, Headphones, Internet connection, etc.

💡 Teaching Tip

There are a lot more inputs and outputs on a smartphone than you students might think of right away. If your students struggle here, consider asking them the following prompts:

- How does the phone know it has to shut down when it is too hot?
- How does the phone know where it is?
- How does the phone know to switch from playing music through the speakers to playing music through the headphones?
- How does the phone know to turn off the screen when you put the phone up to your ear?
- How does Apple know that you got the phone wet?

✔ Assessment Opportunity

Use this wrap up activity to assess how well students have understood the role of input and output in some common activities on a computer. For example:

- Typing on a Keyboard (Input) Makes Letters Appear on a Screen (Output)
- Moving a Mouse or Touch Screen (Input) Changes What Appears on the Screen (Output)
- Pressing play on a touchscreen (Input) Makes a Song Play through the Speakers (Output)

If you need, give students this or other examples to prompt more examples

---

If you are interested in licensing Code.org materials for commercial purposes,**contact us.**

Lesson 6: Processing

Unplugged

Overview

Students complete two unplugged card sorting activities to explore the meaning of processing and its relationship to problem-solving. The first activity has few constraints and is used to introduce a high-level definition of processing. The next introduces more constraints that force students to develop an algorithm that will always successfully process the cards. Students iteratively develop, test, and share their algorithms with classmates. A wrap-up discussion has students reflect on the different types of problem-solving they used in these activities and the value of producing an algorithm to solve a problem.

Purpose

This lesson serves multiple roles. It is another opportunity for students to collaboratively problem-solve. The main activity is a challenging problem that emphasizes the importance of testing solutions on multiple inputs, breaking down large problems, and iterative improvement. In sharing their solutions students see that there are many equally valid solutions to the same problem. These are features students will see in many of the problems they will encounter through the course.

This lesson also introduces the concepts of processing and algorithms. The definitions of processing and algorithms presented in this lesson are intentionally high-level. Students should understand that processing is whatever a computer does to turn inputs to outputs, and that using a computer to process information requires developing an algorithm that reliably does so. The goal of this lesson is NOT for students to understand and compare different classic sorting algorithms, or even necessarily to formalize their own.

Assessment Opportunities

1. **Define processing as the work done (possibly by a computer) to turn an input into an output**

   In the warm up, students should identify the actual card sorting as the processing in the activity.

2. **Define an algorithm as the series of commands a computer uses to process information**

   In the last discussion question, ensure that students identify algorithms as necessary for computers to solve information problems, or to process information.

View on Code Studio

Objectives

Students will be able to:

- Define processing as the work done (possibly by a computer) to turn an input into an output
- Define an algorithm as the series of commands a computer uses to process information
- Develop and iteratively improve an algorithm for processing information based on given constraints

Preparation

☐ Prepare one deck of cards for every 8 students, removing face cards. Pairs will need roughly 8 cards each.

☐ Alternately, print and cut out the number cards. Pairs will need roughly 8 cards each.

☐ Prepare one copy of the activity guide for each student

Links

> **Heads Up!** Please make a copy of any documents you plan to share with students.

For the Teacher
- **Card Sorting** - Exemplar
- **Number Cards** [Make a Copy ▾]

For the Students
- **Card Sorting** - Activity Guide
[Make a Copy ▾]

Vocabulary

- **Algorithm** - A precise sequence of instructions for processes that can be executed by a computer

3. **Develop and iteratively improve an algorithm for processing information based on given constraints**

   On the second page of the worksheet, check that students have improved their algorithms for sorting the cards in the box labeled "iterate".

Agenda

Warm Up (10 min)

   Sorting Cards

Activity (30 mins)

   Card Sorting Algorithms

Wrap Up (10 min)

   Solving Problems with Computers

Teaching Guide

Warm Up (10 min)

Sorting Cards

🎤 *Remarks*

> So far we've explored what inputs and outputs mean in the context of a computer. Today we're going to look more closely at processing by doing a couple of unplugged activities.

Group: Place students into pairs.

💡 **Distribute:** Give each pair roughly 10 cards. Place the cards in a stack face down between pairs. Ask students not to touch the cards until you say so.

Announce the rules of the challenge:

- No talking during the challenge
- The goal is to get the cards in a line, face up, in number order
- When you are done, a team member should say "Done!" and the teacher will give the team a time

Once the class understands the activity and is ready, begin the challenge. You might choose to project a stopwatch if you like for comparison. Once groups have finished give them another opportunity to improve their times.

⊘ **Prompt:** Ask students to silently journal about the following prompts:

- What are the input, output, and processing in this activity?
- How is your approach to solving this problem different from how a computer might have to approach it?

Discuss: Give students an opportunity to talk at their tables about the prompts. Once they are ready to share ask a couple of groups to share their responses to each question.

🎤 *Remarks*

> There are many ways your solutions might be different from how a computer would solve this problem. In general, however, the biggest difference is that computers need to be programmed with a plan without knowing the exact cards that they are going to get. In the next activity we're going to look at how a computer might solve this same problem.

Activity (30 mins)

Card Sorting Algorithms

✎ *Activity Guide - Sorting Cards*

> **Distribute:** Give each student a copy of the activity guide to each student (or pair of students).
>
> **Review Instructions:** Have students read the activity instructions, then review as a class.

Circulate: Walk around the room and listen to the types of ideas students are discussing. Encourage students to practice using the Tips section on the activity guide. Reinforce the idea that they should have a strategy that works for every arrangement of the cards, not just the ones that they have out currently. Ensure pairs are switching between Sorter and Pointer roles.

Share and Test: After several minutes have groups share the approaches they are using. The aim here is to give students a chance to hear how others are approaching the problem. Ask students to actually walk through their methods with the other group. The activity guide provides some possible test-cases that students could consider.

Iterate: Based on what they observed from other groups, ask groups to update, revise, or improve their own method. They should be aiming to create algorithms that work for greater numbers of cards (ideally any number of cards) and that work regardless of the original organization of the cards.

Circulate: Have students return to the challenge, encouraging them to move to eight cards if they haven't already. Once teams have an method they think works, have them write or draw a description of it on the activity guide in the space provided.

Processing Uses Algorithms: Read this section together as a class, calling out the new vocabulary word algorithm.

Model the Rules: You will likely need to model the rules of the activity. For unplugged activities like this one, misconceptions can easily arise. Leave time early in the activity for questions to avoid students spending time on the wrong problem.

💡 Teaching Tip

Speed is Less Important: In the warm up activity the timing can help with motivation. In this activity the focus is much more on more deliberate problem solving. So long as their solutions work, considerations of speed or efficiency aren't important for the goals of this lesson.

Start with Fewer Cards: Encourage the class to start with just three or four cards before trying to test their algorithms on eight or more.

💡 Teaching Tip

What's Success Look Like?: To achieve the learning objectives of this lesson, students don't necessarily need to develop an method that works for eight cards, or even one that always works. If students have an method that is working for fewer cards or only works sometimes, they still will have grappled with the different type of problem-solving required in this activity and will be able to participate in the debrief discussion.

Wrap Up (10 min)

Solving Problems with Computers

Prompt: Either as a discussion or silent journal ask students to consider the following prompt, also found on their activity guides.

- 💬 What step of the problem-solving process did you think was most important in this activity? Why?

- ✅ Why would someone create an algorithm if they already know how to solve a problem by hand?

Discuss: Have students share their responses with a partner and then share as a whole class.

🎙 *Remarks*

Computers help us solve problems by processing inputs to make outputs. As we saw in the second activity today, that can often mean more work up front to develop very specific steps a computer could use to process the information. It can be tricky and require lots of our problem-solving skills to do this. The benefit, however, is that once you have the problem solved once you can make a computer solve that problem for you every time after that.

💬 Discussion

Goal: The first discussion helps contextualize this lesson as another problem-solving activity and provides an opportunity to review the importance of each step. Focus discussion on the "Why" part of the question.

✅ Assessment Opportunity

The second discussion should highlight that while figuring out an algorithm is usually more time-consuming, it means you could then give the instructions to a computer to do that task for you. For example, with an algorithm a computer could sort a million cards, or a million piles of cards, faster than humans and without getting bored. It is a different type of problem-solving that emphasizes eventually automating the solution, and algorithms are necessary for a computer to solve problems.

Standards Alignment

CSTA K-12 Computer Science Standards (2017)

- ▶ **AP** - Algorithms & Programming
- ▶ **CS** - Computing Systems

Lesson 7: Apps and Storage

Unplugged

Overview

This lesson reviews the input, output, storage, and processing aspects of a computer in a context that is relevant and familiar to students: apps. In pairs, students evaluate smartphone applications to analyze the specific problems that they were designed to solve, the inputs that they need to work, and the processing that turns those inputs into the desired output, and what information they would want to store for later. The class concludes with a discussion that connects the lesson to apps students are more familiar with.

Purpose

In Chapter 1 of this unit, students learned the problem solving process. In Chapter 2, students learned how computers solve problems. At this point, students know that computers are information processing machines that can do four things with information: input, output, store, and process. In this final lesson before the unit project, students look at types of input that may be needed to solve a particular problem and describe the processing and storage that a computer would do to produce the desired output. This should prepare them to eventually design their own app to address a problem and explain how that app would work.

Assessment Opportunities

1. **Describe how information can be processed to solve a particular problem.**

   On pages 2 and 4 of the activity guide, students explain how to process information to solve a problem. Check to see that their answers are reasonable. Note that the answers in the exemplar may not be the only correct solution to the problem.

2. **Identify a possible source of a given input.**

   The middle columns of the charts in the activity guide ask students to find a source for various inputs. Check that the answers are reasonable given the type of information, using the exemplar for guidance.

3. **Determine what information should be stored on a device for later.**

   The last prompt in the main activity asks students to think of data that should or should not be stored on a smartphone. Prompt students to defend their reasoning about what types of data should

View on Code Studio

Objectives

Students will be able to:

- Describe how information can be processed to solve a particular problem.
- Identify a possible source of a given input.
- Determine what information should be stored on a device for later.

Preparation

☐ Print a copy of the activity guide for each student

Links

> **Heads Up!** Please make a copy of any documents you plan to share with students.

For the Teacher
- **App Exploration** - Exemplar
- **App Exploration - Sample App** - Presentation

For the Students
- **App Exploration** - Activity Guide
  [Make a Copy ▾]

be stored on a phone, making sure that they are choosing information that will not change frequently and that the program is likely to need later.

Agenda

Warm Up (5 min)

    Solving Problems with Computers

Activity (40 min)

    App Exploration

Wrap Up (10 min)

    Sharing Findings

Extension Activities

    App Store Exploration

Teaching Guide

Warm Up (5 min)

Solving Problems with Computers

Review: Quickly review the input, storage, processing, and output model of a computer

Display: Display the photo of the translation app in**App Exploration - Sample App - Presentation**, or use an app that you think would be relevant to students.

💬 **Prompt:** What problem does this piece of software address? What role do input, output, storage, and processing play in this app?

Allow students to reflect individually before sharing in small groups.

🎤 *Remarks*

> Our sample app used inputs from the camera to solve a common problem. Smartphones have lots of different ways to get input, and we'll be looking at several of them in the next activity.

Prompt: What other types of input can a smartphone use?

Write the student responses on the board. It's not necessary that students produce every possible type of input, but make sure that they understand that input can come from the Internet (such as a list of restaurants in the area), from direct user input (such as pressing a button or filling in a form), and from phone sensors (such as the GPS or gyroscope).

Activity (40 min)

App Exploration

🎤 *Remarks*

> Today you'll be working in groups to figure out how a computer (in this case, a smartphone) uses information to solve problems. You'll be acting as the software in processing the information you get from the inputs, and determining the output that you want to store and to communicate to the user, just as the translation software processed the Spanish text and displayed the English text as output to the user.

Group: Put students in groups of 2-3

💡 **Distribute:** the activity guide.

> Review the instructions for the Ring Silencer Challenge as a class.
>
> **Prompt:** Most of the problem has been defined for us, but we still need to think about what types of output the app will have.

💬 Allow students to discuss with their partners, then share their ideas with the group.

> **Circulate:** Support students as they work in pairs through the first challenge, including the improved version of the app. If students finish a challenge early, encourage them to think of other improvements they

could make to the app.

💬 **Discuss:** What did this app need to know, and what was the output to the user? What about the improved app? Were there any changes to the input you needed? What needed to change about the program?

Circulate: Allow students to work on Challenge 2, supporting them as they complete both the initial challenge and the improved app.

Share: For the last challenge, what inputs did you identify? What sort of processing did you need to do on the information to determine the output? What extra inputs did you need for the improved version?

✓ **Prompt:** For these two challenges, you've used inputs, outputs, and processing, but you also had the opportunity to store information. Is there any information that you think your phone should store? Why? What types of information are generally stored on a smartphone?

Share: Allow students to share out their responses, and write them on the board.

Wrap Up (10 min)

Sharing Findings

💬 **Discuss**: Students share their answers to the questions, and compare the differences between them.

🎤 *Remarks*

Most of the apps that we rely on in everyday life, ones that give us directions or recommend restaurants in the area, fairly open ended. That means that there are many different outputs that could be considered correct, and many different ways that the apps could use the inputs they have available to them.

Prompt: Now, take a few minutes to think of an app that you think is useful, then imagine a way that it could be improved. Share your thoughts with your elbow partner, and work together to think of what extra input you might need to make those improvements work.

🎤 *Remarks*

You'll have a chance to try out some of your ideas as we look to our unit project in which you will create a prototype of an app.

Extension Activities

App Store Exploration

Have students visit an app store like Google Play or Apple's App Store. Instruct them to find a non-gaming app and conduct the same analysis as in the activity guide (problem it solves, information it needs, output it provides to the user).

Standards Alignment

CSTA K-12 Computer Science Standards (2017)

▶ **AP** - Algorithms & Programming

▶ **IC** - Impacts of Computing

Lesson 8: Project - Propose an App

Unplugged | Project

Overview

To conclude their study of the problem solving process and the input/output/store/process model of a computer, students will propose an app designed to solve a real world problem. This project will be completed across multiple days and will result in students creating a poster highlighting the features of their app that they will present to their classmates. A project guide provides step by step instructions for students and helps them organize their thoughts. The project is designed to be completed in pairs though it can be completed individually.

Purpose

This project combines the two major themes of Unit 1, problem solving, and the input-output-store-process model of a computer. This project ties both themes to a broader goal of identifying real world problems and finding ways to use technology to help solve them. Students collaborate in the creation of their app and also take part in a structured peer review process that they will see many more times throughout the course. This project should be a fun and creative experience as well, and gives students a chance to tie the field of computer science to their own interests and ambitions.

Assessment Opportunities

Use the project rubric attached to this lesson to assess student mastery of learning goals of this unit. You may also choose to assign the post-project test through code studio.

Agenda

Warm Up (10 min)
> Introduce the Project

Activity (150 min)
> Project Guide

Wrap Up
> Presenting Apps

Extended Learning
> Shark Tank

Post-Project Test

View on Code Studio

Objectives

Students will be able to:

- Identify and define a problem that could be solved using computing
- Design an app that inputs, outputs, stores, and processes information in order to solve a problem
- Provide and incorporate targeted peer feedback to improve a computing artifact

Preparation

☐ Print a copy of **Apps and Problem Solving - Project Guide** for each pair of students

☐ Poster paper, pens, markers and other supplies for making posters

Links

> **Heads Up!** Please make a copy of any documents you plan to share with students.

For the Teacher

- **Apps and Problem Solving** - Project Guide Exemplar
- **Apps and Problem Solving** - Sample Marked Rubrics

For the Students

- **Apps and Problem Solving** - Project Guide
  [Make a Copy ▾]
- **Apps and Problem Solving** - Rubric
  [Make a Copy ▾]
- **Apps and Problem Solving** - Peer Review
  [Make a Copy ▾]
- **Computer Science Practices** - Reflection
  [Make a Copy ▾]

Teaching Guide

Warm Up (10 min)

Introduce the Project

Say: This whole unit we've been learning about the problem solving process and how computers process information to help solve problems. In the previous lesson you learned how apps help solve problems by taking inputs and processing them to create useful outputs. Today we're going to start a project where you and a partner will design an app to solve a problem of your choosing.

Distribute: **Apps and Problem Solving - Project Guide** , one copy for each student, as well as **Apps and Problem Solving - Rubric**. As a class review the information provided on the first sheet of the project guide which explains the project, lists the steps, and shows what students will need to produce. Then provide a brief overview of each of the more detailed steps.

Activity (150 min)

Project Guide

Step 1: Choose a Partner : Place students in pairs or groups of three.

Step 2: Brainstorm Problems: Prompt groups to spend several minutes silently brainstorming problems and recording them on their project guides. Circulate the room and remind them that at this point they shouldn't be thinking about an app they want to build or even whether it's possible to solve this problem with an app. Make sure they're beginning with the problem rather than the solution.

Step 3: Choose Your Problem : The project guide provides several criteria students can use to assess which of their problems they'd like to address. Ask students to look forward to Step 4 if they need more guidance on how they'll need to define or scope their problems. Give students a few minutes to discuss with their group and choose the problem they'd like to

Step 4: Define Your Problem : For this step students will need to appropriately scope their problem by defining who their audience is, what specifically is the problem, and how they will know they have fixed it.

Step 5: Your App : Once students have scoped their problem ask them to discuss an app that could be used to help solve their problem. To begin they'll just need to provide a high level description of the app that describes how a user would use it and what it does.

Step 6: Input, Output, Store, Process : In this step students design the way their app will actually work to process data.

> 💡 Teaching Tip
>
> **Creating Groups:** Ideally this project is done in pairs. If need be, groups of 3 will work. You should decide beforehand whether you will assign or allow students to pick their partners.

> 💡 Teaching Tip
>
> **Helping the Brainstorm:** Listen carefully to student conversations. Identify students who are stuck, and the reason why they are stuck. Are they thinking too big? Help those students to think about problems as annoyances or inconveniences, or an opportunity to improve your quality of life to a small or big degree.

> 💬 Discussion
>
> **How Much to Help:** At this point students have had a lot of practice defining problems. They also have a peer review process shortly after this step. Encourage them to be as detailed as possible but avoid giving specific advice for how to define their problem.

First students will draw and then describe the outputs of their apps. On the left side they can make a rough sketch of what their app would look like. This does not need to be a final draft and is just there to help them brainstorm ideas and communicate to another group how their app would look. On the right side they have space to label what each individual piece of information on the screen.

Using the outputs that students selected as a guide, students should pick the inputs they'll need to create them. Only 6 spaces are provided though students could opt to choose more. This is somewhat intentional to help students scope the functionality of their app.

Students will describe the way their app processes data using as a model the way they would process it themselves. The goal here is primarily just to ensure students have selected inputs that could be processed to produce the outputs. For example, if they're finding a list of friends with birthday this month then both a computer and human would need to know a list of friends' birthdays and the current month.

Lastly students are asked to decide what information, if any, it makes sense to store long term.

Distribute: Apps and Problem Solving - Peer Review , one copy to each pair of students

Step 7: Peer Review: Each group should trade their project guides with another. They should fill out the first line of the peer review which asks what specific part of their project they'd like feedback on. Afterwards there are a number of directed questions as well as a chance to provide more open-ended feedback on the idea.

Students should be given their project guides back as well as their peer feedback. On the back there are questions where they can indicate what changes or improvements to their projects they intend to make on their apps based on the feedback.

Step 8: Finalize App and Make Poster : Students should incorporate the ideas of their peers in finalizing their app idea. They should then make a poster presenting their app following the guidance provided on the activity guide.

Wrap Up

Presenting Apps

Share: Decide if and how students will share their posters with one another. If students will be doing more formal presentations then use the guidelines provided in Step 9 of the project guide to structure the presentations.

Collect: At the end of presentation collect the completed project guides, peer feedback forms, and posters from each group.

Reflect: Students reflect on the following question in their journals:

- What are you most proud of in your project?

🖥 Send students to Code Studio to complete their reflection on their attitudes toward computer science. Although their answers are anonymous, the aggregated data will be available to you once at least five students have completed the survey.

🖥 Code Studio levels

- Levels
- ✅ ☰ 2

Student Instructions

View on Code Studio ⤢

Extended Learning

Shark Tank

Run a mock "Shark Tank" as the backdrop for this unit project. Some things to consider:

- Is the culture of your class one where this competition can remain at a healthy level?
- Invite faculty / staff, local residents, or other professionals to hear the Phase 2 presentations and decide on a first, second, third place idea.
- Invite local business people / other professionals to share during Day 1 or Day 2 so students can see how computer science relates to jobs in their community.

Post-Project Test

The post-project test is found at the bottom of the Problem Solving and Computing unit overview page on Code Studio (**studio.code.org/s/csd1-2019**).

This test is locked and hidden from student view by default. In order for students to see and take this test, you'll need to unlock it by clicking the "Lock Settings" button and following the instructions that appear.

Standards Alignment

CSTA K-12 Computer Science Standards (2017)

- ▶ **AP** - Algorithms & Programming
- ▶ **CS** - Computing Systems