Lesson Plan: Algorithms in Maine-based Businesses
(for Unit 2: Problem Solving)

OVERVIEW

Purpose of this lesson

These video lessons are meant to show students that:

a. computational thinking is part of everyday jobs in their local area, and
b. not all "computer science" related jobs involve programming.

They are also meant to support one or more of the Computational Practices listed in the ECS curriculum (from p. 12 in ver. 7). Which computational practices best align will depend on the approach each teacher takes with the lesson, so once you determine your approach, you can emphasize the relevant practices with students.

ECS Computational Practices

- **Analyze the effects of developments in computing (impact/connections)**
- **Design and implement creative solutions and artifacts**
- **Apply abstractions and models**
- **Analyze their computational work and the work of others**
- **Communicate computational thought processes, procedures, and results to others**
- **Collaborate with peers on computing activities**

This particular lesson illustrates the use of algorithms in an automotive repair shop in Damariscotta, ME.

We suggest you use this video as an introduction to Unit 2, or early in it, to emphasize (a) one or more of the computational practices listed above and which are being encompassed in jobs all around the state, and (b) that appropriate uses of computers are part of computer science.

ECS connections

This short lesson supports the following objectives introduced in ECS Unit 2: Problem Solving (ver. 7, p. 22):

- Summarize the behavior of an algorithm.
- Explain the characteristics of problems that cannot be solved by an algorithm.

This work is supported by National Science Foundation grant #1440464

Lesson Plan: Algorithms in Maine-based Businesses
(for Unit 2: Problem Solving)

ACTIVITY INSTRUCTIONS

Engage

Have students do one or more of the following.

1. In small groups and/or as a whole class, have students discuss their current understanding of algorithms† (step-by-step instructions). Consider using questions like the following.

    a. *An algorithm is a process or set of rules to be followed during calculations or other problem-solving operations, with or without a computer.*
    b. Describe an "algorithm" or "process" you've followed, without the aid of a computer, in order to solve some sort of problem. *(Challenge students to come up with non-school-based examples from their "real" lives.)*
    c. Describe a time you solved a problem of some sort where a computer device performed a sequence of steps for you.
    d. How do you think an auto mechanic goes about figuring out what is wrong with your car? How would a checklist be useful? How is a checklist like an algorithm? How is it different?
    e. Can you think of jobs where algorithms are important? (If students offer only computer-based jobs, challenge them to think of ways people use algorithms without computers.)

2. Have students use a dichotomous key (e.g.- for plant or animal identification) and discuss how the key works and what type of thinking goes into creating one.

Explore/Explain

Have students do one or more of the following.

1. Show students the Auto Mechanic video about algorithms (https://youtu.be/OdciJW6Oj0Q). Help them connect this to the *Engage* activity they did. For example:

    a. Look for problem-solving steps in the video.
    b. What role does a dichotomous key play in this setting?
    c. How are algorithms used in the business shown in this video?
    d. What are the limitations of the algorithms illustrated in this video?

    After the video, have students brainstorm other local jobs that likely make use of algorithms (step-by-step instructions) either at a low or high tech level.

2. Have students conduct local interviews with people who somehow utilize algorithms (step-by-step instructions) in their work, either at a low or high tech level. Have students report back to the class about the experience (e.g.- class discussion, create a slideshow, write a

† At what point one introduces the term *algorithm* will depend on the readiness of one's students.

This work is supported by National Science Foundation grant #1440464

story or new report, etc.) Sample interview questions might include:

a. We are learning about "algorithms," which is the idea of doing something by telling a computer a set of steps to do, and then giving it different kinds of data and seeing what it comes up with after it does the steps.
b. If you think of all the different things you might do in your day, can you think of anything you do where you use a computer to do a series of steps for you?

Sample Interview Questions

1. **Can you tell me a bit about how it works?**
2. **Where does the algorithm come from/ who created it?**
3. **What is the result of using this algorithm, and what do you do with this result?**
4. **What does a human have to do and what does the computer do?**
5. **Why is it useful to have a computer do this algorithm instead of you having to do it on your own?**
6. **Are there any instances where the algorithm can't or doesn't give you a good result?**

3. Have students create their own video-based interview of a local use of algorithms. *(See sample interview questions under option 2.)*

Explain/Elaborate

Have students do one or more of the following.

1. After showing the auto mechanic video, conduct another discussion to address questions such as the following.

    a. Think of an algorithm you heard Darryl mention in the video.
        i. What makes it an algorithm?
        ii. What were some of its steps?
    b. Why aren't the checklists (algorithms) the auto mechanics use "generic" or "one-size-fits-all"?
    c. What are some limitations of algorithms such as those Darryl describes in the video? How do the mechanics deal with these limitations?
    d. To what extent is it possible for these flowcharts to be automated by a computer, so that the mechanics would just do one test after another without the need to think about what they're doing?

2. Have students create their own dichotomous key to identify something (e.g.- people in the room, everyone's shoes).

This work is supported by National Science Foundation grant #1440464

3. Have students write their own algorithm for some specific task or skill, such as sharpening a chainsaw, making a cake, etc.‡
4. If during the prior stage students conducted interviews (video-based or otherwise) they could now vote on which was the most interesting and plan a field trip to the location. Have them plan in advance what questions they would ask during the visit.
5. If students watched the auto mechanic video during the Explore/Explain stage, they could elaborate on that experience now by creating their own video-based interview of a local use of algorithms.
6. Conduct a general discussion about algorithms to help students identify how their thinking about them has changed. Discussion questions might include:

    a. What other types of jobs might use algorithms, and for what purpose?
    b. What can algorithms not solve? What are the limitations of algorithms?
    c. What surprised you?

Evaluate

We have not included any evaluation since teachers have many directions they can take with this lesson, but suggest that an evaluation or assessment for formative purposes will be useful.

‡ During the Human-Computer Interactions unit students write instructions for making a peanut butter and jelly sandwich. It could be interesting to have students reflect on how they improve each time they write out such algorithms/instructions.

This work is supported by National Science Foundation grant #1440464