Concurrency Means Cooperation

Provided by TryEngineering - www.tryengineering.org

Lesson Focus

This lesson provides a number of kinesthetic exercises that illustrate how teamwork can contribute to efficient problem solutions. The lesson includes practice in figuring out how to divide up a problem, and reassemble it. Students also explore how scientists use the Internet and idle computing power to do calculations on volunteer machines. If possible, with sufficient teacher expertise, students set up a computer to contribute to solving such a problem.

Age Levels

Recommended for 8 – 10

Objectives

Introduce students to:

- Sequential computing.
- Concurrency in computing.
- What can go wrong, such as deadlocks.

Anticipated Learner Outcomes

Students will be able to describe:

- What concurrency is.
- How concurrency works.
- How a large scientific project can be shared through the Internet.

Alignment to Curriculum Frameworks

See attached curriculum alignment sheet.

Internet Connections

- http://folding.stanford.edu/home/
- https://boinc.berkeley.edu/projects.php
- https://boinc.berkeley.edu/index.php
- https://youtu.be/myomEBjnIDw

Recommended Reading

- http://resources.sei.cmu.edu/asset files/CurriculumModule/1990_007_001_15815.pdf
- https://en.wikipedia.org/wiki/Actor_model
- https://en.wikipedia.org/wiki/Deadlock
- https://en.wikipedia.org/wiki/Parallel_computing

Optional Writing Activity

- Explain what concurrency is and give an example of a problem that can be solved concurrently.

Concurrency Means Cooperation

For Teachers:

◆ Lesson Objectives

Students will describe:

- What concurrency is.
- How it works.
- How a large scientific project can be shared through the Internet.

◆ Materials

- Access to the Internet to do Internet research.
- A lot of paper to write instructions lists.
- Pencils.
- A bowl of any size for every five students in your class.
- A small paper plate (or napkin) for each student.
- Chopsticks sufficient for each student to have one chopstick (not a pair, just one).
- Something edible in the bowl that meets the dietary constraints of your school but can be picked up with chopsticks, for example wrapped candy.

◆ Procedure

Videos on concurrency appropriate for middle schools are almost non-existent. The few available are more than 10 minutes and will cut into active learning time. Therefore, both of the following sessions provide an opportunity for students to explore concepts and vocabulary through kinesthetic games and Internet search.

In Session 1, students will do a bit of theater to explore how concurrent systems can do a task with a single data resource. They will also use a concurrent model to collect information to select a volunteer computing problem in which to participate. In Session 2, students will connect their computer to a volunteer grid. They will also simulate the 'dining philosophers problem' and explore ways that concurrency can cause problems.

To illustrate the actor model, you will use your creativity to set the students on a task that will help you. Cleaning your classroom is a good one. For example, you might need to sort the markers from the colored pencils, clean out all the desks, return books to their proper places on the shelves, pick things up off the floor, water the plants, feed the fish. (Your colleagues may want to do this lesson as well!)

Make a list of the major tasks, then create instructions for someone else to do that task, listing the steps needed for completion. For example one task might be to clean the students' desks, with detailed instructions on what to keep, what to put away elsewhere, and what to throw out. Each of the subtask instructions should be expanded into a detailed list a separate pages. Another task might be to separate the markers from the colored pencils from the regular pencils, from the pens. Again, the instructions list itemizes 'Collect all the markers', 'Collect all the pens', etc.

In other words, you are making lists that expand into other lists that may expand into other lists. Three or four levels deep will work best. You will only need a few copies of the top-level instructions (e.g. the clean the room task) list, but you may need one copy per student for the 'clean a student desk' instruction sheet. Make sure each instruction sheet

has a place for the name of the person who did the job, and an item that says 'Get help if you need it.'

Have a reward handy for your students when you finish this. You will be giving rewards based on jobs completed. Make sure everyone gets a chance to complete reasonable jobs. They are, after all, cleaning your classroom for you.

Session 1:

1. Explain to your students that the traditional description of a computer algorithm is that it is like a 'To do' list. Show them the 'To do' list to clean the classroom. Explain that if you had to do this yourself it would take a long time. But by using concurrency they can help you get the job done. Explain that for each task they are assigned and satisfactorily complete, they will get a point. Points convert into rewards.
2. Explain the definitions of concurrency and the actor model to them. (These appear on the Student Resource sheet.) Emphasize the idea that every actor can make his or her own decisions about how to split up the subtasks for the task he or she was assigned. Illustrate this by giving a student the task to 'clean the student desks'. Talk the class through the idea that this actor can assign the task 'clean a desk' to each student in the room, or to a group of tidy, speedy students, or repeatedly give the job to one other actor. Remind this student that the more time it takes for the subtasks to be completed, the longer it will be before she can report that her task is done.
3. Once everyone understands the rules, set yours students loose by giving a student the instructions for each of your subtasks. They in turn have to distribute tasks to others. There will be controlled bedlam. It is within your rights as the initial actor to pause or halt the system and do a partial or full restart. Try to let them finish out the tasks, and leave 20 minutes for the Internet search task (if you can do it during the session), and another 10 for a wrap up discussion.
4. Concurrency can be used to accumulate information. The Student Resource page has two concepts defined. Depending on the Internet resources in your room (e.g. only a teacher station, a few computers, tablets, smart phones, etc.), or your access to a computer lab, you can do the actual Internet research in this session or assign it for homework. Alternatively, meet as a computer lab during the second session and do the actual research then. Give each student the task of completing the resource page. Ask them how they could negotiate to distribute the work amongst themselves so that 1) they know they have the right answer, and 2) everyone gets the job done.
5. Have a short discussion about the experience of concurrent tasks, what worked and what didn't work. Did everyone share the load? Would it have helped to have someone in charge, telling everyone what to do?

Session 2:

1. If you didn't complete the Internet search for concepts activity, do it now. Focus on the discussion of volunteer and grid computing. As a group, discuss whether there are particular projects your students would like to participate in. Develop an instruction sheet with a list of how your class could get involved in a particular project either with school computers or at home. (Many of the projects now run on smart phones and tablets.) If there is enthusiasm and a well-constructed instruction list, perhaps you can pursue this after this session.

2. Break your class into as many groups of five as you can, then create a group to distribute the remaining students into the groups of five. Have students in the groups of six take turns observing and recording the activity, but let everyone sit at the table. Distribute the worksheet, along with one chopstick per person, and a bowl of edibles. Help the students set up the table as instructed on the worksheet. Review the rules, and reinforce that they may only grab food with two chopsticks, and they must use the chopsticks to their left and right (e.g. share them). They cannot share across the table, and no fingers in the bowl.
3. As they eat, allow them to negotiate how everyone can get fed fairly. Remind them that the must use two chopsticks. Remind them to notice whether they experienced deadlock or starvation. Stop the action when the food is gone, or whenever you think they get the idea.
4. Wrap up by reviewing the Student Resource sheet and making plans to do volunteer computing.

◆ Time Needed

- ✦ 2 sessions, at most 1 hour each. This can be done in a single two-hour session if your students are typically cooperative during group activities.

Concurrency Means Cooperation

Student Resource: (You fill out the rest as part of a concurrent activity)

- **Concurrency** In computer science, is a property of systems in which several computations are executing simultaneously, and potentially interacting with each other.

- **An actor** is a computational entity that, in response to a message it receives, can concurrently:
  - Send a finite number of messages to other actors
  - Create a finite number of new actors
  - Designate the behavior to be used for the next message it receives.

- **Parallel computing** is

- **Distributed computing** is

- **Deadlock** is

- **Starvation in concurrent programming** is

- **Volunteer computing** is

- **Grid computing** is

- **A Volunteer or Grid Computing Activity we would like to do** is

Concurrency Means Cooperation

Student Worksheet: Dining Philosophers

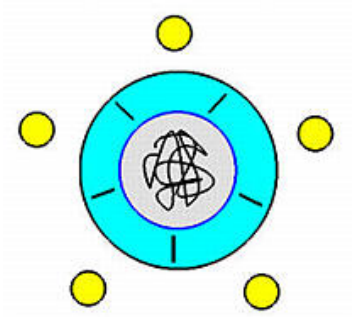
An old concurrency puzzle involves five philosophers who eat together and think. That's all they do. They sit around a table with one bowl of something delicious to share. They use two chopsticks to pick up what is in the bowl. The problem is that there are only five chopsticks among them.

Simulate the dining philosophers' problem. You will need a bowl of food (no fingers in it please), five chopsticks, and five bowls or napkins to place in front of each philosopher. Create the table setting at a table, use a student desk, or sit on the floor. (Any food that falls on the floor goes into the garbage.) In the illustration below, the grey circle is the bowl, lines in the blue circle are chopsticks, yellow circles are plates or napkins. Image from https://www.cs.mtu.edu/~shene/NSF-3/e-Book/MUTEX/TM-example-philos-1.html.

Here are the rules:

- You must have a chopstick in each hand to grab food from the bowl. You will have to figure out how to share.
- If you reach deadlock or starvation, whoever calls it out first gets to pick food from the bowl.
- You may stop the action at any time to negotiate how to fairly share the food.

If you have an observer, that person should take notes for the class discussion. You can also use the observer to act as a timer if that helps you share fairly.



Concurrency Means Cooperation

For Teachers:

Alignment to Curriculum Frameworks

Note: All lesson plans in this series are aligned to the Computer Science Teachers Association K-12 Computer Science Standards, and if applicable also the U.S. Common Core State Standards for Mathematics, the U.S. National Council of Teachers of Mathematics' Principles and Standards for School Mathematics, the International Technology Education Association's Standards for Technological Literacy, the U.S. National Science Education Standards and the U.S. Next Generation Science Standards.

◆National Science Education Standards Grades K-4 (ages 4-9)

CONTENT STANDARD E: Science and Technology

As a result of activities, all students should develop

- Understanding about science and technology

◆National Science Education Standards Grades 5-8 (ages 10-14)

CONTENT STANDARD E: Science and Technology

As a result of activities, all students should develop

- Understandings about science and technology

◆Next Generation Science Standards & Practices Grades 3-5 (ages 8-11)

Practice 2: Generating and Using Models

- Use a model to test cause and effect relationships or interactions concerning the functioning of a natural or designed system.

Practice 6: Constructing Explanations and Designing Solutions

- Use evidence (e.g., measurements, observations, patterns) to construct or support an explanation or design a solution to a problem.

◆ Standards for Technological Literacy - all ages

Nature of Technology

- Standard 2: Students will develop an understanding of the core concepts of technology

The Designed World

- Standard 17: Students will develop an understanding of and be able to select and use information and communication technologies

◆CSTA K-12 Computer Science Standards Grades 3-6 (ages 8-11)

5.1 Level 1: Computer Science and Me (L1)

- Computational Thinking (CT)

  2. Develop a simple understanding of an algorithm (e.g., search, sequence of events, or sorting) using computer-free exercises.

- Collaboration (CL)

  3. Identify ways that teamwork and collaboration can support problem solving and innovation.