Outreach-in-a-Box: Discovering IT

Set-the-Stage and Extension Activities

Dear Educator:

Set the stage for the upcoming IT-in-a-Box presentation by completing one or both of these 20-minute activities. Each one accesses prior knowledge and prepares students for the learning ahead. Prior to her or his visit, please review the completed activity with the presenter, and discuss ways to bridge from this activity to the guest presentation. Follow the class visit with Extension Activities that take the learning further.

Set-the-Stage Activities

Set-the-Stage Activity One: Who's Behind the Technologies We Use?

- Description:

- Objective:

Access prior knowledge about IT and its importance in daily life, and build awareness of technology as something people create.

- Directions:

1) Introduce the presentation to come. You may wish to say: We have a special guest coming from (institution) to teach us about computer science and information technologies. During the hour we will learn about the world of computing and do some fun activities that will get you thinking about opportunities ahead. To get started, let's tap into what we already know and like about technology.

2) Have students contribute to a list of all the computing devices they can think of. Say: If it is programmable (you can control complex functions) it probably has a computer chip in it and it's a computer!

List a few devices on the board or chart pad to get juices flowing: iPod, cell phone, control panel on a microwave oven, a DVR like TiVo. Now record items as students contribute. If they are not sure an item is a computing device (which is OK!) just put a question mark beside the item. As kids run dry, suggest different settings, like a car, a hospital, the 911 dispatch office, a grocery store.

3) When you have a big list (20 or 30) ask students to write a reflection about the following: Choose any one of these devices. Think about who might have been behind its development. Imagine: What motivated them?

4) After five minutes, ask students to stop and look up. Say: Let's see what you picked for your reflection. As I run my finger down the list, raise your hand when I get to the item you wrote about. Look around – which ones seemed to interest people? Why might that be?

5) Ask students to share several reflections and discuss. Choose items that several students chose in order to compare their thinking, and also pick way-out-there items students might have interesting insights about.

6) Bring the point home, saying: Every computing device we use was built intentionally to solve a problem or improve on an earlier invention. Now imagine YOU get to design the next new device or software. Take a minute and write about your invention and what motivates you to create it.

7) After five minutes, ask students to share their ideas in triads; then have each triad share one idea with the class. Congratulate students on creative thinking.

8) Ask students to prepare for the upcoming visit by listing a set of questions they would like to ask the presenter. Our upcoming guest makes her [or his] living in IT and will tell us more about the different kinds of work people do in the field. We'll want to ask what motivated her [or him] to be in a technical field. Write this on

chart paper or the board: What motivated you to do the work you do? Ask students what other questions they would like the presenter to answer, and add them to the list.

9) In advance of the upcoming program, share the reflections and questions with the presenter. Discuss ways she or he might bridge from this activity to the presentation.

Set-the-Stage Activity Two: Computer Technologies in the World

- *Objective:*

Access prior knowledge and awareness of living in a digital world.

- *Directions:*

1) If you have not done so already, introduce the presentation to come. You may say: We have a special guest coming from (institution) to teach us about computer science and information technologies. During the hour we will learn about the world of computing and do some fun activities that will get you thinking about opportunities ahead. To get started, let's tap into what we already know, or what we THINK we know.

2) Have students meet in small groups of four or five to brainstorm applications of computer technology in the home. Write this on the board for reference, and say: Computer technologies are everywhere, and there are a lot of them in our homes.

Brainstorm a list of all the computer technologies at work in modern homes. Challenge teams to work for ten minutes to make an exhaustive list. If they are unsure whether an application involves computing, have them write it anyway. As they work, encourage them to think of every room in the house, from the bathroom to the kitchen to the garage.

3) After ten minutes, reconvene and ask teams to take turns reading their lists. Record all responses on a running list on poster paper. As in the game Boggle, have each team cross out any item that appears on all teams' lists. Along the way, teams may challenge whether an application involves computing. Record disputed applications as well, but put a question mark next to them. (Encourage students to pursue their own research to define what a computer is and whether the disputed applications involve computing, and remind them they can learn more from the presenter who will visit.) At the end, the team with the most unique and undisputed items wins the challenge.

4) Discuss how this exercise would differ if teams thought about a different setting – a grocery store, an animation studio, a car, a medical research lab, and so on. Ask students to brainstorm all the places where digital technologies are likely to be prominent, and write these on another list. Ask interested students to research these settings for all the computer technologies at work in them and report back. (Example, grocery store: bar code scanners, deli weigh scales with label printers, climate control, security cameras and alarm system, inventory control systems, digital design on packaging, employee 'clock in' stations, cash registers, back-office computers and printers).

5) Ask students to prepare for the upcoming visit by listing a set of questions they want to ask the presenter. Collect these and share with the presenter in advance of the upcoming program. Discuss how she or he can follow up, answering questions, discussing disputed items from the game, defining what a computer is versus another electrical device.

6) Have all work available and visible as a jumping off point when the presenter arrives.

Extension Activities

Extensions include: Programming activities, robotics, and careers exploration. The amount of time each takes depends on the activity, the number of students involved, and the depth you go. You may wish to test the activities in advance, or walk through them with the hosting teacher.

Note: If you cannot complete these additional activities, please share them with the hosting teacher. He or she may wish to use them to extend your program.

Extension Activity One:

Tell A "Robot" How To Make A Cream Cheese & Jelly Sandwich

- *Description:*

The presenter acts as a robot and follows instructions from students in order to build a cream cheese and jelly sandwich. Since the "robot" is a mechanized computing device that only understands finite and sequential instructions and makes assumptions about nothing, it executes the instructions just as they are given, with funny and illuminating results.

- *Objective:*

Students understand that computing devices function by acting on precise and sequenced instructions, and these instructions are delivered through computer program languages that devices "understand" and execute.

- *Materials needed for activity:*

Raised flat surface to work on, loaf of sliced bread, jar of jelly, package of cream cheese, plate, butter knife, spoon.

- *Directions:*

Introduce the activity. You may wish to say:

We are going to do an activity that helps you understand what is at the heart of computing – programming. All computing devices function by acting on precise and sequenced instructions. These instructions are delivered through computer program languages that devices "understand" and execute.

In order to think about how robots or any computing devices are programmed to do the things we want them to do, I'm going to pretend to be a robot – a computer with mechanical functions - that needs to make a cream cheese and jelly sandwich. I'll take my instructions from you, one step at a time. In front of me is a loaf of bread, jar of jelly, a package of cream cheese, plate, butter knife, and a spoon.

In order to encourage balanced participation, you may want to choose students by counting off every third student or moving up and down the rows for each next instruction. Or, you may ask the teacher to call on students for you so you can stay in robot mode. Remember, encourage inclusive participation and don't let "tech" kids dominate.

As students deliver instructions avoid taking any assumed steps. If you're told by a student to "put the cream cheese on the bread," pick up the package of cream cheese and place it on the bread. If a student says, "Spread the cream cheese on the bread," but the cream cheese is still in its packaging or you have not been told to pick up the knife, you can say that you are unable to complete that task at this time (or something similar). Wait for a student to tell you to put the knife into the cream cheese package and get some cream cheese on it before executing a spreading "command." When the bread bag is open and you have one slice of bread on the plate, if you are told to do something with the bread, ask which bread. The key is to get students thinking about how to properly sequence the actions needed while providing precise detail for you, the "robot."

Try not to give any analysis of the students' commands during the activity. Let students use the opportunity to figure out on their own how to issue instructions to the "robot."

End the activity whenever it seems the point has been made even if you have not constructed a complete sandwich. Conclude the activity and discuss what happened. Call on a variety of students to answer (even those not raising hands). Some sample questions:

What surprised you about what we just did? (Answers will vary)

What things can you assume when giving instructions to a human that you can't when giving them to a computer or "robot"? (Humans have ears and eyes, can interpret voice commands and visual cues. Robots need a computer chip that transmits instructions and sensors that interpret what they are acting upon. Humans make assumptions and fill in gaps. Computers do not.)

What if you could only use a restricted set of commands? Which ones do you think would be the most useful?

When you told me to pick up the bread, I did. Do you think it would be easy for a robot to find the bread? (Could lead to a discussion of how a robot "sees," with sensors).

How could you have told me to move if I said I didn't know what bread was?

Discuss robotics as an important science. Ask: Are robots just for fun? Most robots do real work. If a job is boring or dangerous, a robot is probably doing it! When might robots be most useful? (Repetitive tasks like making the same weld on each car moving down an assembly line, dangerous tasks like bomb removal, or working with hazardous chemicals.)

Close the discussion by telling students the robot activity was a programming exercise, and remind them programming is at the heart of all computing. To make a computing device do anything it needs precise instructions, and these are written in computer languages they can learn. The brochure you hand out has resources that they can use to explore computing and robots right away.

Extension Activity Two: Tell a Computer to Draw a Picture

This is a Kinesthetic Learning Activity (KLA), meant to physically engage students in the learning process. KLAs fill an important niche — energizing students, employing underutilized learning styles, and achieving especially challenging learning goals. Discuss this activity with the teacher and plan how students will pair, arrange themselves, and use their materials.

- *Description:*

A student "programmer" thinks about transmitting information so a fellow student "computer" acts on it, moving his or her pencil to draw a line drawing that only the "programmer" sees. Students may describe using some terms that express scale and position, but they will find it takes precise instructions for the drawing to come close to matching the original picture.

- *Objective:*

Students understand that computing devices function by acting on precise and sequenced instructions, and these instructions are delivered through computer program languages that devices "understand" and execute.

- *Materials:*

Enough photocopies of several different line drawings so each pair has one sheet (samples included); 8.5" x 11" sheets of blank paper; pencils or pens, space in the room to have students sit back to back on the floor or at tables or desks with a divider between them (could use a propped up 3-ring binder).

- *Directions:*

Introduce the activity. You may wish to say:

In order to think about how you could program a robot or other computing device to draw a picture, we're going to do a drawing activity in pairs. One of you will be the "programmer" giving instructions and the other will be the computing device acting on those instructions and drawing. You should sit such that neither of you will be able to see the other's paper. You can do this by sitting back to back or by propping a 3-ring binder or large book on the table between you. (You may wish to ask the teacher in advance which method is best.)

Say: I'll start by giving a line drawing to each programmer. [It is important that students sitting near each other have different drawings, otherwise they can listen to descriptions given by describers on other teams.] If you are the programmer, don't tell the "device" what the picture is. Instead, try to tell your partner how to draw the object in simpler terms, describing pieces of the drawing one at a time. The "device" cannot ask any questions, but may ask for a command to be repeated if they can't act on it. After the picture has been described and drawn, wait for my cue and then you can show your pages to one another. This activity should only take five minutes or so.

After drawing is complete ask pairs to raise their hands if their two pictures (the original and the drawing) ended up different from one another. Call on pairs to describe how the drawings differed (e.g., item placement, shape, scale)

Ask pairs: What terms did you use to describe the task? Ask if other groups used other terms.

Discuss how computers need to have precise language. For example, it would not be enough to say, "Draw a square." The size of the square would need to be specified. Computers (think: "compute") use mathematical terms to express position, shape, size, and relationships.

Select one original and hold up all its renderings so students can compare. Ask: Which picture most closely matches the original? Ask that pair of students what kinds of terms they used. Ask the students if they could think of other ways to describe drawings to a computer. If no one has thought about describing line segments in terms of end points, ask if that might be a good way to describe drawings. Other alternatives to suggest:

• Pen up, pen down, north for ‹distance›, south for ‹distance›, west for ‹distance›, and east for ‹distance›. Ask the students if any of them have drawings that could not have been done (easily) with this method (anyone with a circle or diagonal line should show their drawing).

• Drawing a grid, then moving left to right across each row, saying if a grid square should be colored in or not. (Can lead to a discussion of pixels on a computer screen.)

Ask: Do you think it would have helped you to know what the object was that you were trying to draw? Do you think it would help a computer?

If time allows, switch roles of programmer and computer and repeat the activity with different pictures. At the conclusion, ask students if they did anything differently this second time. Did they change their terms after the discussion about different approaches? How? If so, did they have a better result?

Close the discussion with a suggestion that students try their hand at programming and programmable devices like robots. The brochure and links to the NCWIT Web site offer many opportunities, including activities, camps, devices to buy, competitions, do-it-yourself crafts, and more.

Conclusion (5 minutes): If your hour presentation ends here, take a moment to discuss the brochure and insert. Encourage students to explore IT!

Extension Activity Three: Programming

Scratch and Alice are two programming languages suitable for youth. Each Web site below has information for teaching.

Scratch

Scratch is a programmable toolkit that enables kids to create their own games, animated stories, and interactive art -- and share their creations with one another over the Internet. Scratch is being developed by the Lifelong Kindergarten research group at the MIT Media Lab, in collaboration with KIDS research group at the UCLA Graduate School of Education & Information Studies. The Scratch Web site supports adult use of Scratch with kids: http://weblogs.media.mit.edu/llk/scratch/educators.html

Alice

Middle school to college-age students learn to program interactive 3D graphics with Alice v2.0. Alice is an object-oriented, Java-based computer-programming environment created by Carnegie Mellon University researchers. The Alice Web site and Alice Community Forum support the use of Alice with kids. http://www.alice.org/

Extension Activity Four: More Robotics

If you have the means to bring Crickets, Lego Mindstorms, a Roomba vacuum or other robotic devices, consider these steps for an inquiry activity.

1. Simply put the device to work and ask: What do you think this is? What is it responding to? How does it sense the world? How would you describe its "brain"? What makes it go?

2. Explain that a robot is a mechanical computer. It functions through a processor, responds to programming instructions, and interacts with the world with sensors. Robots are fun but they are useful. If a work task is incredibly repetitive or hazardous, a robot is probably performing it.

3. Show students how you change one instruction to make the device function differently. Describe the other things it can do (PicoCrickets can "dance," twirl, and jump).

4. Have students form small teams to plan a secret function they want the robot to perform. They will get three tries to make it perform as desired. As each group interacts with the computer, the rest of the class watches, and asks questions about what they are trying to make it do, and offers suggestions for changing its behavior.

5. Discuss the ways in which working with the robot is similar to the "programming" activity they did prior.

6. Answer questions about the devices.

7. Remind students their takeaway brochure has information about "virtual" robotics on the Web, and robots they can buy.

Extension Activity Five: University of Washington Careers Videos

Show one or more of three videos from University of Washington Computer Science & Engineering so students can see people working and talking about their work in IT. All videos from Why Choose CSE? University of Washington, www.cs.washington.edu/WhyCSE.

• Power to Change the World – First-person accounts of computer science and engineering students, alumni, and faculty explaining why they chose computer science as their field. Use the video to introduce computing as an exciting field full of opportunities.

• Pathways in Computer Science – Illustrates the diverse professional pathways students can pursue after receiving a degree in computer science or computer engineering. Use the video to explore how a degree in computing prepares students for almost any imaginable future.

• A Day in the Life – Six brief profiles of recent computer science and engineering graduates. Meet bright young women engaged in secure, highly collaborative, creative, diverse, challenging, and well-compensated work. These role models will resonate with young people who might not otherwise consider a career in computing.

Following the videos, use these steps:

1. Have students write a reflection on this prompt: Use your imagination - If you could change the world and use technology to do it, what would you do? What would you invent or improve? Encourage students to be imaginative and have fun, and tell them there are no right or wrong answers.

2. Have students share their reflections and discuss how their ideas might tie back to fields of CS and IT.

3. Encourage students to keep their eyes open, talk to people who do what interests them, and to follow their passion!