Tiny Tech Lesson Plan

DIRECTIONS/SEQUENCE

Gather: All share names in circle.

Discussion: What is Tiny Tech? Some technology comes with directions (step-by-step instructions on how to put it together). Examples: new toys, bicycles. Some technology comes with the directions already inside of it (step-by-step instructions on how it should work). Example: computers, smartphones, iPads. SOMEONE, a person, had to write out those directions (in a special language) to tell the computer what to do; they had to "program" or "code" the computer; tell it what to do step-by-step. We are here together to learn the first steps in telling a computer what to do, to make you little programmers!

Now "directions" can also mean which way to go—Left, Right, Up, Down.

Story: Go Dog Go by P.D. Eastman

Discussion: Discuss Left, Right, Up, Down directions

Game: Simon Says – use commands to review Left, Right, Up, Down; can also use students' names in individual commands to learn names

Story: The Very Hungry Caterpillar by Eric Carle (Discuss sequence of events)

Discussion: Sequence – a particular order in which things follow each other. There is an order to how we do things, certain steps we have to follow for everything to turn out right. Has anyone ever eaten a bowl of cereal? What if your mom poured the milk BEFORE she put the bowl out?! How do you prepare a bowl of cereal? (discuss the steps) Also, counting is a sequence (demonstrate how counting on one hand only ends at 5 if you count in order).

Game: Correct Sequence…
Organize stories into the correct sequence. Each student gets two "stories" to organize. Can share one with class or pass.

Discussion: As a programmer, you have to tell your computer what order to do things in or it will get confused and not work. A computer will do what you want exactly in the order you tell it to, so you have to make sure to give commands in the right order (In Kodable you are directing a little fuzzball out of a maze. It won't figure out itself to go up if

Tiny Tech Lesson Plan

you've told it to go another direction). You have to lead it out of the maze by giving it directions in the right sequence/the right order.

Discussion: tablet is the "technology" we will use; discuss rules for using iPads
- Rules: no dropping
- Hold with two hands when walking
- Swipe/Touch
- If problem, tell me – don't punch screen or punch buttons

Action: Pass out iPads now, start Kodable; start 1st level (1.1.1) together, continue through Smeeborg Sequence levels and stop when reach debugging levels

To close: if some children finish before others, they can play Kodable games (hit left/back arrow until main screen, select Vocabulary to play mini-games)

Tiny Tech Lesson Plan

Class 2
DEBUGGING

Story: Click Clack Moo Cows That Type by Doreen Cronin

Discussion: Sometimes we have plans that don't work out and we need to change our plans. Sometimes we are trying one way to solve a problem that is not working, so we have to try another way to solve that problem. What was the cow's plan? Was it working? How did they change their plan? And did it work then? We call this process problem solving. In computer programming, it's called DEBUGGING. When your program isn't running correctly, you need to look back through your program, find the bug (the mistake), and try something different.

Review: Words learned in previous weeks
Sequence – a particular order in which things follow each other

Activity: Pull up Kodable curriculum Unplugged Activity Algorithms and complete Fuzz Family FaceOff together as group.

Discussion: In some of today's Kodable levels, a program will already be written, but there is at least one mistake in it, maybe 2 or 3. Students need to follow the program to see where the mistake is and DEBUG the program by correcting the mistake.

Action: Hand out iPads with Kodable started and student names selected. Students will start where they left off last week and hopefully get through last 5 levels of Smeeborg Sequence which are debugging levels (1.1.11-15).

Tiny Tech Lesson Plan

CONDITIONS

Story: The Runaway Bunny by Margaret Wise Brown (emphasize IF…THEN)

Discussion: Condition – what needs to happen before something else is allowed to happen. If…then. Have any of you ever heard your parent say, "IF you eat all your dinner, THEN you can have dessert?" or "IF you clean up your room, THEN we can go play?" Those are conditions. You have to do one thing first before you can do something else. One thing must be true before some other thing happens.

When your parents reward you like that, what is the condition of the reward? Just like your parents will check on your room to see if you cleaned it, a computer will also check on a statement to see if it is true to decide which action to take.

Sometimes we want to have an extra condition, in case the 'if' statement is not true. This extra condition is called an 'else' statement. IF you clean up your room, THEN you can play on the computer, ELSE your brother will get to pay on the computer. If it is true you clean up your room, you get to play on the computer. If it is false that your room is clean, your brother gets the computer.

Game: Conditionals Cards – As an example, let's play with this deck of cards. IF I draw a 7, everybody claps. ELSE everyone says 'Awwwww.' Draw card and see if class reacts appropriately. Ask: What was the IF? What was the ELSE? Which condition was met? Repeat.

Now create a few 'programs' with your class that depend on things like a card's suit, color, or value to award points. Write the program on whiteboard as an algorithm:
IF (card is red)
    award teacher 1 point
ELSE
    award class 1 point
Run one program at a time (using about 10 cards or 5 points as ending point). Draw cards and follow the program to see how many points each team scores in each round. Play several times with several different programs to help the students really understand conditionals.

Tiny Tech Lesson Plan

Discussion: Now sometimes in a computer program there can be several ways to do things (example: in a game, if you press A, then your guy will jump; if you press B, then your guy will duck); there are different conditions to tell your computer about. With our fuzzballs, there may be more than one way to go, and we have to give him directions for the best way to go. Sometimes there are conditions in our fuzzball maze.

Action: Instructor starts Kodable; shows students how to do level 1.2.1 and 1.2.2 while explaining conditions (coins will indicate the BEST way to program your fuzzball, not the only way; IF there is a purple tile, THEN go down). Pass out iPads. Students start Kodable, do own level 1.2.1, continue at own pace through level 1.2.15, stop when reach third section (Loopy Lesson)

Tiny Tech Lesson Plan

Class 4
LOOPS

Story: We're Going on a Bear Hunt by Michael Rosen and Helen Oxenbury

Discussion: Repetition – doing or saying something over and over. What phrases repeat in this story? If you took out all the repeated phrases in these stories, there would be a lot less story, a lot less words. That might not make for a great story, but in a computer program, using less code/less words is important. Using less code saves time for the programmer and saves time for the computer (and we all like our computers to work quickly, right?). When repeating things, sometimes we can see a pattern.

Activity: Complete the Pattern
As a whole, class will call out pattern on worksheet as instructor points to it. Demonstrate how you can loop the patterns; place () around the whole pattern and have students count the number of times that pattern repeats on the worksheet.

Discussion: In our mazes today, we are going to look for patterns that repeat. When we find them, we will put those repeated patterns into a loop so we can give the fuzzball less commands with the same result.

Game: Loop the Pattern…
Have students stand up in a line. Give them each a direction to point their arms with a repeated pattern (right, up, right, up, right, up, right, up)(right, down, right, down, right, down, right, down) Direct the "repeats" to move in front of the original set (forming a T) to simulate the loop command in Kodable. Then have the "repeats" sit down while counting them to see how many times the loop needs to repeat.

Action: Instructor demonstrates on paper how to complete the first maze using all the commands listed out, then shows how to condense the patterns into a loop function to see how much shorter the code is. Demonstrate 1.3.1, 1.3.2 together having the students help find the patterns and repetitions to loop. Pass out iPads. Students start Kodable, do own level 1.3.1, continue at own pace through loops levels.

Tiny Tech Lesson Plan

Class 5
IMAGINATION

Story: Oh The Thinks You Can Think by Dr. Suess

Discussion: Imagination – Programmers must use imagination to have the computer solve
        problems and to make fun games for us to play – a computer can open the door to
        different worlds and different languages and different ways of thinking about things

Review: Go over key concepts from previous weeks
        Sequence – a particular order in which things follow each other
        Debugging – figuring out what's not working in your plan and fixing it
        Condition – what needs to happen before something else is allowed to happen (If…then)
        Repetition – doing or saying something over and over = Loops

Action: Need to remember all these lessons as we enter the imaginative world of the Foos.
        Students will progress as far as they can using the Foos app. Instructor will use keywords
        when assisting students to help review programming concepts.