ECT Lesson Plan: Writing a Story

Lesson plan at a glance...

Core subject(s)	English/Language Arts; Computer Science
Subject area(s)	Language; Software Development Fundamentals
Suggested age	10 to 14 years old
Prerequisites	*None*
Time	**Preparation:** 8 to 15 minutes **Instruction:** 85 to 105 minutes
Standards	**Core Subject:** CCSS ELA **CS:** CSTA, UK, Australia

In this lesson plan…

Lesson Overview

In this lesson, students act as storytellers by writing chapters of a story collaboratively. Each member writes one chapter independently that starts and ends with fixed story points. After all the members have written their individual chapters, the small group collaborates and makes adjustments to each chapter (**decomposition**) so that the entire story becomes logical and cohesive. The process of melding the chapters involves **pattern recognition, abstraction and generalization**. Students analyze the process to generalize how the collaboration process could be more effective and describe how they would revise the plan for another time (**algorithm design**).

Materials and Equipment

- □ For the student:
  - ○ *Required:* Journal
    - ■ If using Google Docs (http://docs.google.com) or a wiki
      - Internet-connected computer (one (1) computer per student recommended)
    - ■ If not using a computer-based collaboration tool
      - Markers/Whiteboard or Paper and Pen/Pencil

Preparation Tasks

	If students are using computers, confirm that all students' computers are turned on, logged-in, and connected to the Internet	3 to 5 minutes
	Write the story points Create a set of n + 1 story points for your students (a group of 4 members needs 5 story points). An example set of story points for a story with 4 chapters: • Story Point 1 (starting point for chapter 1): "How are you doing today?" said Mom. • Story Point 2 (ends chapter 1 / begins chapter 2): Finally, I finished my breakfast and walked to the bus stop. • Story Point 3 (ends chapter 2 / begins chapter 3): It was just not a very good day for me. • Story Point 4 (ends chapter 3 / begins chapter 4): I was happy when I heard what my teacher said about my project. • Story Point 5 (ends chapter 4 and the entire story): It was nice that I got rewarded for my hard work.	5 to 10 minutes

The Lesson

Warm-up Activity: Point of view	5 to 15 minutes
Activity 1: Writing the story chapters	30 to 45 minutes
Activity 2: Creating a logical and cohesive story	30 minutes
Wrap-up Activity: Analysis and reflection	15 minutes

Warm-up Activity: Point of view (5 to 15 minutes)

Activity Overview: In this activity, students identify or explore the concept of 'point of view.'

Activity:
Choose one of the following student activities to start the lesson. Each warm-up activity is designed to prepare your students for the lesson.

Instruct the students to answer the question **"What might be hard or easy about writing a story with someone else when you can't talk to each other first?"** in one of two ways, encouraging them to consider how point of view is important when writing creatively:

1. Journaling: Students respond in their journal or word processor:

2. 4-S Brainstorming:

- Four students will be selected by the teacher to play a specific role in helping the rest of the class generate the maximum number of responses to the question.
    - *Accelerator* - Student encourages classmates to generate more ideas ("Let's get more ideas, only two minutes left")
    - *Acceptor* - Student helps classmates commit to an idea ("All ideas are OK, write that one down")
    - *Exaggerator* - Student encourages classmates to generate different kinds of ideas ("We need some silly ideas")
    - *Connector/recorder* - Student makes the connection between different ideas and writes them down ("Which ideas are connected to that?")

Activity 1: Writing the story chapters (30 to 45 minutes)

Activity Overview: In this activity, every student will independently write one assigned chapter of a story.

Notes to the Teacher:
Divide the class into groups with the number of group members corresponding to how many chapters are in the story. Assign each member of a group one chapter of the story. If there are 6 story points for 5 chapters, each group needs 5 students.

Activity:
- Each student writes their own chapter **without** consulting or sharing with the other students.
- Each chapter must advance the story by starting from the previous story point and ending at the next story point.

The first chapter starts with Story Point 1 and ends with story point 2. Chapter 2 starts with Story Point 2 and ends with story point 3, and so on. The length of each chapter can be decided based upon the skill level of each student group.

Activity 2: Creating a logical and cohesive story (30 minutes)

Activity Overview: In this activity, students will combine their separate chapters into a single story. The chapters might be radically different, requiring students to **decompose** the story points, point of view, and overall story, and **recognize patterns** in the chapters to create one cohesive story.

Notes to the Teacher:
Students will take turns reading their individual story chapters in order and then collaborate to make the story logical and cohesive. Let your students know that it is normal and expected that at the chapters will not make sense in the context of each other until they start to adjust the inconsistencies.

Activity:

1. Everybody in the group reads all of the chapters for that group, paying attention to the differences.
2. The group picks one chapter as the anchor. They will make minimal changes to that chapter.
3. The group revises the other chapters so that they logically fit with the anchor chapter.

While revising chapters, try to keep each chapter as unique as possible. The final goal is to make the entire story more cohesive by blending each chapter, not to rewrite everyone's chapters.

Wrap-up Activity: Analysis and reflection (15 minutes)

Activity Overview: Students discuss what went wrong with creating their stories and how the task could have been organized to make those problems less likely through **algorithm design**, **pattern recognition**, and **pattern abstraction and generalization**.

Activity:
Discuss the following questions with students in a large group:

- Do you know what **debugging** is? What was the most difficult part of "debugging" your story? Did entire chapters need to be adjusted, or could you manage to reconcile the chapters by making only minor changes?
- What would you have changed about your initial storytelling process to make the debugging process easier? Would you have made your story more straightforward and logical, or more ridiculous and imaginative? Essentially, how would you go about writing the story so that it includes the fewest number of "bugs"? (**algorithm design**)
- How would you change the rules of this activity to guarantee that a minimal number of "bugs" are created? Assume that all chapters must still be written independently by different students at the same time.
- Identifying "bugs" is recognizing patterns. Can you describe some of the common bug patterns that could be useful for other writers to use to debug their story? This is relevant to **pattern recognition**, and **pattern abstraction and generalization**.

Assessment:
Have students journal an answer to this formative question, and evaluate based on the depth and honesty of their answers:
Individually, reflect on this activity. Describe how collaboration makes a task easier. Describe the challenges that collaboration can create. From what you learned in this activity, how would you plan or organize the next collaborative project you participate in?

Learning Objectives and Standards

Learning Objectives	**Standards**
LO1: Students will be able to analyze the task of working on a project with others when they are unable to communicate with the other team members.	*Computer Science* CSTA L2.CT.15: Provide examples of interdisciplinary applications of computational thinking.
LO2: Students will be able to write a sequential story line for a chapter given a starting point and an end point without consulting others.	*Computer Science* CSTA L2.CT.7: Represent data in a variety of ways including text, sounds, pictures and numbers. UK 3.6: Understand how instructions are stored and executed within a computer system; understand how data of various types (including text, sounds and pictures) can be represented and manipulated digitally, in the form of binary digits. *Common Core* CCSS.ELA-LITERACY.W.5.3/6.3/7.3/8.3/9-10.3 Write narratives to develop real or imagined experiences or events using effective technique, descriptive details, and clear event sequences.
LO3: Students will be able to work collaboratively to create a cohesive story from the individually written chapters.	*Computer Science* AUSTRALIA (Creating digital solutions by: defining): Define problems in terms of data and functional requirements, and identify features similar to previously solved problems CSTA L2.CT.1: Use the basic steps in algorithmic problem-solving to design solutions (e.g., problem statement and exploration, examination of sample instances, design, implementing a solution, testing and evaluation). UK 3.2: Understand several key algorithms that reflect computational thinking [for example, ones for sorting and searching]; use logical reasoning to compare the utility of alternative algorithms for the same problem. *Common Core* CCSS.ELA-LITERACY.W.5.3/6.3/7.3/8.3/9-10.3 Write narratives to develop real or imagined experiences or events using effective technique, descriptive details, and clear event sequences. CCSS.ELA-LITERACY.W.5.4/6.4/7.4/8.4/9-10.4 Produce clear and coherent writing in which the development and organization are appropriate to task, purpose, and audience. (Grade-specific expectations for writing types are defined in standards 1-3 above.)
LO4: Students will be able to analyze the process and suggest strategies that would have led to a more successful process.	*Computer Science* AUSTRALIA 6.6 (Creating digital solutions by: designing): Design, modify and follow simple algorithms represented diagrammatically and in English involving sequences of steps, branching, and iteration (repetition)

	CSTA L2.CT.6: Describe and analyze a sequence of instructions being followed (e.g., describe a character's behavior in a video game as driven by rules and algorithms).

Additional Information and Resources

Lesson Vocabulary

Term	Definition	For Additional Information
Point of View	The perspective (or type of personal or non-personal "lens") through which a story is communicated (e.g. first person, third person omniscient)	http://en.wikipedia.org/wiki/Narration
Bug	An error or problem in a process that prevents it from being successful. Typically refers to an error or problem in a computer program.	http://en.wikipedia.org/wiki/Debugging
Debug	To systematically find and remove errors or problems from a process. Typically done to computer programs that do not work correctly.	http://en.wikipedia.org/wiki/Debugging

Computational Thinking Concepts

Concept	Definition
Decomposition	Breaking down data, processes or problems into smaller, manageable parts
Pattern Recognition	Observing patterns and regularities in data
Pattern Generalization	Identifying and extracting relevant information to define main idea(s) while creating models of observed patterns to test predicted outcomes
Algorithm Design	Develop the instructions to solve similar problems and repeat the process

Administrative Details

Contact info	For more info about Exploring Computational Thinking (ECT), visit the ECT website (g.co/exploringCT)
Credits	Developed by the Exploring Computational Thinking team at Google and reviewed by K-12 educators from around the world.
Last updated on	07/02/2015
Copyright info	Except as otherwise noted, the content of this document is licensed under the Creative Commons Attribution 4.0 International License, and code samples are licensed under the Apache 2.0 License.