
ECT Lesson Plan: Divide and Conquer

Lesson plan at a glance...

Core subject(s)	Computer Science
Subject area(s)	Algorithms and Complexity
Suggested age	11 to 14 years old
Prerequisites	None
Time	Preparation: 0 minutes Instruction: 45 to 60 minutes
Standards	Core Subject: CCSS Math CS: CSTA , UK , Australia

In this lesson plan...

- [Lesson Overview](#)
- [Materials and Equipment](#)
- [Preparation Tasks](#)
- [The Lesson](#)
- [Learning Objectives and Standards](#)
- [Additional Information and Resources](#)

Lesson Overview

In this lesson, students will use a 'divide-and-conquer' strategy to solve the mystery of "stolen crystals" using **decomposition** to break the problem into smaller problems and **algorithmic design** to plan a solution strategy.

Materials and Equipment

- For the teacher:
- *Required:* Token (can be anything small but unique: soft toy, favorite pen, trinket, an ID card, etc.)
 - *Recommended:* Stopwatch (online stopwatch - <http://tools.arantius.com/stopwatch>) or clock

Preparation Tasks

	None	0 minutes
--	------	-----------

The Lesson

Warm-up Activity: Searching for the token	15 to 20 minutes
Activity 1: Solve the crime	15 to 20 minutes
Wrap-up Activity: Assessment	15 to 20 minutes

Warm-up Activity: Searching for the token (15 to 20 minutes)

Activity Overview: In this activity, students will use decomposition to discuss how to break down a large search problem into smaller questions with only two answers (yes or no), eliminating possible locations of the missing token or coin. They will use algorithm design to discuss how to find the token in the shortest amount of steps.

Activity:

Display a small token item (a soft toy works well), and announce to the students that you have a telepathic link to the item. To prove your telepathy, you are willing to let the class hide the token with one of the students in the class. A chosen spokesman for the class will answer your yes/no questions. Using the answers, you will find the student holding your precious hidden token.

Pick a 'foreman' to organize the class - the foreman can be the one to hold the token. The foreman is responsible for distributing the token and answering your yes/no questions. Give them 60 seconds to decide who will hold the hidden token, after which time you will begin your questions. Use a stopwatch (or designate a student as timekeeper) to countdown from 60 seconds. Give your token to your foreman, close your eyes and turn around.

After 60 seconds, warn them that you are about to turn around, so all students can be ready to go. After you turn around, walk the room trying to sense an 'aura'; every now and then exclaim that you feel close. After a short while, return to the front. Ask everyone but the foreman to line up in a row.

Ask the two students in the middle of the line to take a step away from each other. Ask the foreman whether the token is in the second 'half' of the line (remind them this is a valid yes/no question if they complain). When you know which half the token is in, ask the other half to sit down. Repeat the same process with this half, until you only have two people and you can ask if one of the two has the token.

After you have retrieved your token, lead a discussion on what other strategies could have been used to find the token.

Is the teacher's strategy efficient?

Is there a more efficient strategy?

How do you measure efficiency?

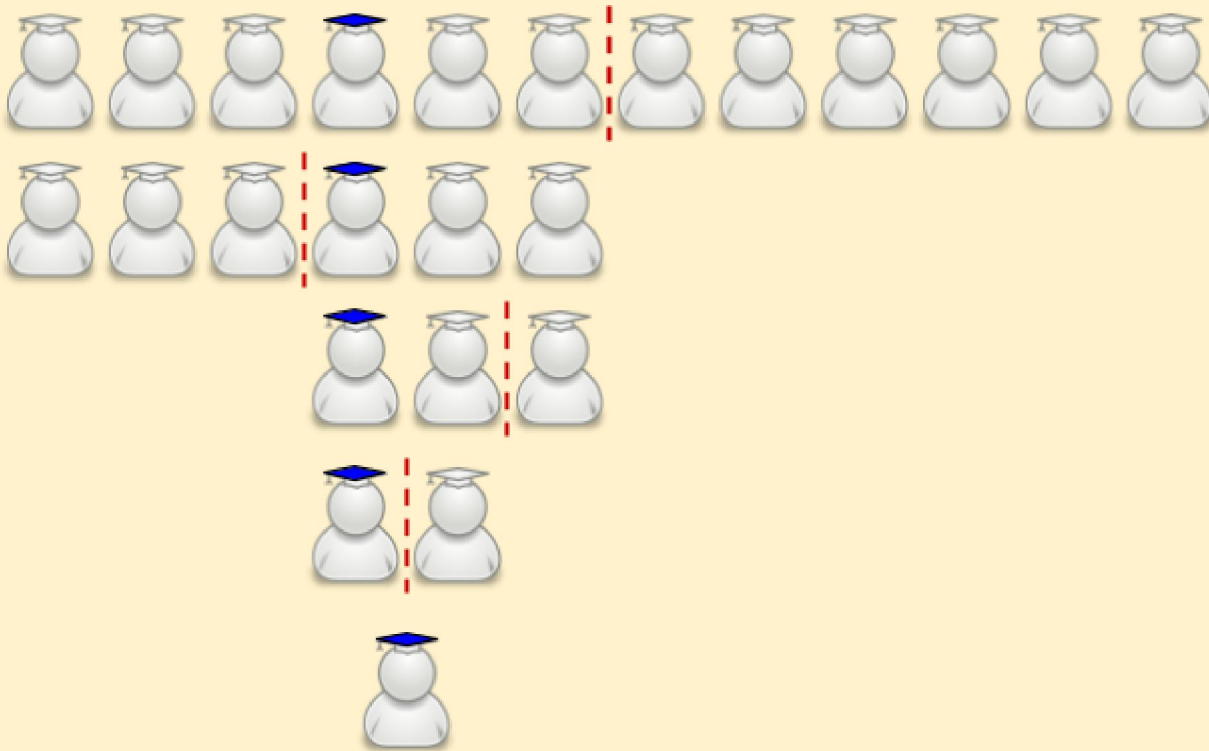
Notes to the teacher:

The number of questions that must be asked to locate the token will depend on the number of students in the class: for n students, the maximum number of questions needed is $\log_2 n$. So for a class of 25 students, you would need $\log_2 25 = 5$ questions. As a quick guide:

Number of Students	Questions Needed
0-2	1
3-4	2
5-8	3
9-16	4
17-32	5
33-64	6

In a class of 12 students for example, we can see from the table above that only 4 questions are needed. By lining up the students, we can repeatedly divide (by asking the question 'is the token in this half of the room?') until we are left

with only one person. The following diagram illustrates this - the blue cap indicates the position of the student with the token, and the dashed red line where we would halve the line:



Assessment:

Q1: What other strategies could have been used to find the token?

A1: You could guess every student, row by row, until you find the right one; you could pick students randomly until you find the right one; you can eliminate entire regions of the classroom by asking students in the corner whether the person who has the token is sitting in their row/column.

Q2: Why was the method the teacher used an efficient one?

A2: Because it halved the size of the problem with each step, so it became easier and easier to solve the problem. By asking every person, you are no nearer solving the problem after each time you incorrectly pick a person.

Activity 1: Solve the crime (15 to 20 minutes)

Activity Overview: In this activity, the scientists must determine the time the crime occurred and the only data they have is a stack of photographs taken every five minutes over the past 24 hours of the place where the eggs were the crime occurred. Students will use decomposition to break down a large time-search problem into smaller questions about whether or not an egg is visible in a single photograph. Students will use algorithm design to choose the best method (binary search) to find the maximum number of time-coded photographs that must be viewed to find the time when crystal eggs were stolen from a laboratory.

Activity:

Explain to the students that the strategy you used to find the hidden token problem is known as 'divide and conquer' - a strategy for solving problems by breaking them into smaller and smaller problems that can be more easily solved. This is a really powerful and useful tool in many different situations.

Announce suddenly that you are thinking of becoming a professional crime solver. Explain that you read about a heist at a laboratory recently, and realized you could solve the problem very quickly. Describe what you know about the story:

1. The scientists noticed a set of crystal eggs had been stolen when they reported to work early Tuesday morning. Reliable witnesses say the eggs were definitely there on Monday morning. The thieves must have come in sometime between Monday morning and Tuesday morning.
2. The camera system had been disconnected, but the laboratory had a backup system: a webcam took a picture of the crystals every 5 minutes and stored it securely. Every hour 12 pictures were taken, so in the 24-hour period, $12 * 24 = 288$ pictures were taken.

Tell the students that you could have solved that mystery after only a few seconds if you were given the pictures.

In collaborative student groups of 3-5, students must determine the maximum number of pictures their team needs to look at in order to determine the time span in which the crime occurred.

Notes to the teacher:

Remind the students that viewing all the pictures from start to finish would take too long. However, by using a divide-and-conquer approach, they only need to look at a handful of pictures before they could find the time at which the crystals were stolen.

With 288 pictures, you can see if the crystal eggs were stolen half-way during the day by checking the 144th picture (144 is half of 288). If the crystal eggs were still in the half-way picture, you know that it was stolen in the second half of the day. When you know in which half of the day the theft occurred, you can check in the middle of that group of 144 pictures to identify the next time break. Continue dividing the pictures in half until you can find the first picture in which the crystal egg is missing:

1. Check the 144th picture to find out which 12-hour period the eggs were stolen in
2. Check the 72nd picture in that group to find out which 6-hour period the eggs were stolen in
3. Check the 36th picture to find out which 3-hour period the eggs were stolen in
4. Check the 18th picture to find out which 90-minute period the eggs were stolen in
5. Etc.

Assessment:

Assess student understanding from the group answers. Discuss the process used to arrive at the correct answer. Invite a student group to demonstrate the solution.

Wrap-up Activity: Assessment (15 to 20 minutes)

Activity Overview: In this activity, students will use decomposition to break down a large search problem into smaller questions with only two answers and use algorithm design to choose the best method to find the answer to a problem.

Activity:

Create another example that is most quickly solved using a divide-and-conquer (binary search) approach. Examples:

- The thieves with the crystal eggs want to give it to their accomplices on the ground while they hide out. What is the highest floor they can hide on and be able to safely drop the eggs to the accomplices below?
- Find the page location of the word “magnanimous” in the dictionary by looking at the fewest number of pages.

Assessment:

Assess individual or group answers of the students at the end of the lesson.

Learning Objectives and Standards

Learning Objectives	Standards
LO1: Students will be able to decompose the problems into smaller challenges.	<p><i>Common Core</i> CCSS.MATH.PRACTICE.MP7: Look for and make use of structure.</p> <p>HS-ETS1-2: Design a solution to a complex real-world problem by breaking it down into smaller, more manageable problems that can be solved through engineering.</p> <p><i>Computer Science</i> CSTA L2.CT.12: Use abstraction to decompose a problem into subproblems.</p>
LO2: Students will be able to collaborate in developing an algorithm for solving a problem.	<p><i>Common Core</i> CCSS.MATH.PRACTICE.MP2: Reason abstractly and quantitatively.</p> <p>CCSS.MATH.PRACTICE.MP1: Make sense of problems and persevere in solving them.</p> <p><i>Computer Science</i> AUSTRALIA 6.4 (Creating digital solutions by: defining): Define problems in terms of data and functional requirements, and identify features similar to previously solved problems.</p> <p>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).</p> <p>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.</p>
LO3: Students will be able to evaluate the efficiency of a problem-solving strategy.	<p><i>Next Generation Science Standards</i> HS-ETS1-3: Evaluate a solution to a complex real-world problem based on prioritized criteria and trade-offs that account for a range of constraints, including cost, safety, reliability, and aesthetics as well as possible social, cultural, and environmental impacts.</p> <p>CCSS.MATH.PRACTICE.MP2: Reason abstractly and quantitatively.</p> <p><i>Computer Science</i> AUSTRALIA 6.4 (Creating digital solutions by: defining)</p> <p>CSTA L2.CT.1</p> <p>UK 3.2</p>
LO4: Students will be able to independently apply the problem solving	<p><i>Common Core</i> CCSS.MATH.PRACTICE.MP2: Reason abstractly and quantitatively.</p>

strategy to a new problem.	<p>CCSS.MATH.PRACTICE.MP1: Make sense of problems and persevere in solving them.</p> <p>CCSS.MATH.PRACTICE.MP5: Use appropriate tools strategically.</p> <p><i>Computer Science</i></p> <p>AUSTRALIA 6.4 (Creating digital solutions by: defining)</p> <p>CSTA L2.CT.1</p> <p>UK 3.2</p>
----------------------------	--

Additional Information and Resources

Lesson Vocabulary

Term	Definition	For Additional Information
Binary Search Algorithm	A series of instructions that finds the position of a specified input value (the search "key") within an array sorted by key value .	http://en.wikipedia.org/wiki/Binary_search_algorithm

Computational Thinking Concepts

Concept	Definition
Algorithm Design	Creating an ordered series of instructions for solving similar problems
Decomposition	Breaking down tasks into smaller, manageable parts

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 .