Algorithms: Tangrams

Lesson time: 20 Minutes   Basic lesson time includes activity only. Introductory and Wrap-Up suggestions can be used to delve deeper when time allows.

LESSON OVERVIEW

This lesson shows us something important about algorithms. If you keep an algorithm simple there are lots of ways to use it. If you want to make sure everyone ends up with the same thing, then your algorithm needs more detail.

This activity will show both options.

TEACHING SUMMARY

Getting Started - 10 minutes

1) [Review](#)
2) [Vocabulary](#)
3) [Intro to Tangrams](#)

Activity: Algorithms with Tangrams - 20 minutes

4) [Algorithms](#)

Wrap-up - 10 minutes

5) [Flash Chat: What did we learn?](#)
6) [Vocab-Shmocab](#)

Assessment - 10 minutes

7) [Algorithm Assessment](#)

LESSON OBJECTIVES

Students will:

- Tackle the challenge of translating an image into actionable instructions
- Convey instructions to teammates in order to reproduce an image
- Analyze the work of teammates to determine whether an outcome was successful

TEACHING GUIDE

MATERIALS, RESOURCES AND PREP

For the Student

- [Tangram Set & Algorithm Card Images Pack](#)
- Scratch paper for writing algorithms or building images

- [Tangram Assessment Worksheet](#)
- Pens/Pencils
- Scissors

For the Teacher
- Print one [Algorithm Card Images Pack](#) per group
- Print one [Tangram Set](#) per student
- Print one [Tangram Assessment Worksheet](#) per student
- Provide student with scissors, paper, pens & pencils

GETTING STARTED (10 MIN)

1) Review

This is a great time to review computer science experiences from the past. Where did everyone leave off?

Here are some questions that you can ask in review:

- Do you remember what an algorithm is?
- What does an algorithm have to do with sequence?

> **LESSON TIP** *Finishing the review by asking about the students' favorite things helps to leave a positive impression of the previous exercise, increasing excitement for the activity that you are about to introduce.*

2) Vocabulary

This lesson has one important word to review:

Algorithm - Say it with me: Al-go-ri-thm
A list of steps that you can follow to finish a task

3) Intro to Tangrams

Your students may or may not have played with tangrams before. If they have, you can skip this portion, and move right to explaining the Algorithms activity.

Usually, Tangrams are used to solve puzzles. You receive a set of seven Tans and must use them all (without overlapping any) to recreate an image that has been given to you. Often, this is done as an individual activity, and the player is allowed to see the image that they are trying to recreate. Many times, you can lay your pieces right on

top of the image silhouette to be sure that the solution is just right.

ACTIVITY: ALGORITHMS (20 MIN)

4) Algorithms

We are going to use our tangrams in a slightly different way than most. Instead of looking at our puzzles and trying to guess which shape goes where, we are going to get puzzles that already tell you where each shape goes.

You might think that this will make it easier, but it won't, because students will also not get to actually *look* at the image that we are trying to recreate! Instead, a teammate will be *describing* the image to us.

To keep it from getting too difficult, we will not use puzzles that require all seven pieces.

Directions:

1. Divide into groups of 3-5.
2. Each player should cut out their own set of tangrams.
3. Have one member of each group select an Algorithm Card without showing it to anyone else.
4. The person with the Algorithm Card will try to explain the image to everyone else without letting them actually see it.
5. The other players will build their pictures off of the description given by the Card Holder.
6. When the Card Holder is done, everyone will show their pictures and see if they all ended up with the same image.
7. If everyone ends up with the same drawing, the Card Holder can show the card and see if everyone matched the card.
8. If any of the pictures in the group are different from each other, have the Card Holder try describing the image again, using more detail.
9. Choose a new Card Holder and a new Algorithm Card and repeat until everyone has had a chance to describe an image.

Play through this several times, with images of increasing difficulty.
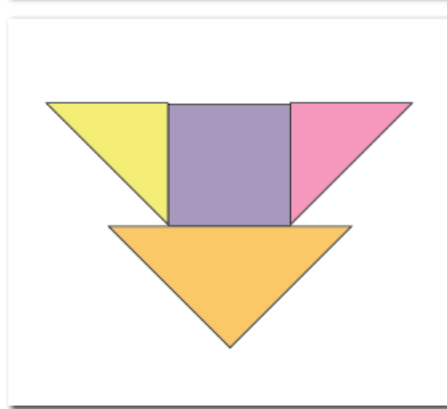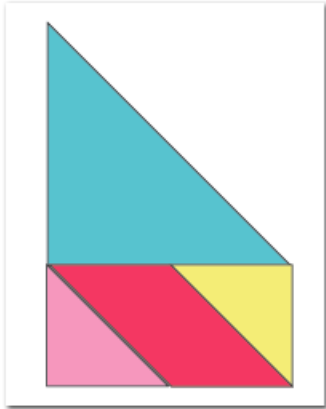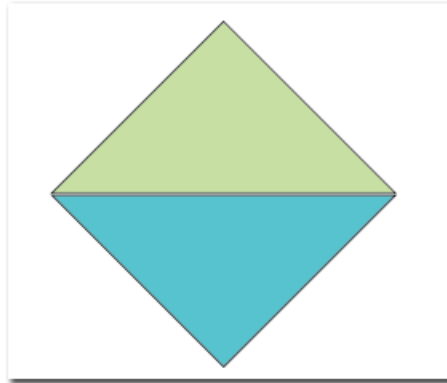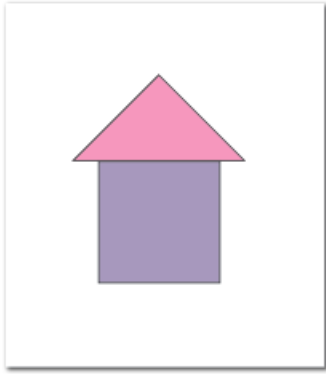
WRAP-UP (10 MIN)

5) Flash Chat: What did we learn?

- What did we learn today?
- Was it easier or harder than you thought it would be to describe an image to one another?
- Did any group end up having arrangements that all matched?
- Can you share some tricks that you came up with that helped your group match the Image Card exactly?

6) Vocab Shmocab

You can choose to do these as a class, or have the students discuss with an elbow partner and share.

- Do you remember the definition of the word "algorithm"?

    "A list of steps that you can follow to finish a task"
    "An algorithm that has been coded into something that can be run by a machine"
    "Finding and fixing problems in your algorithm or program"

ASSESSMENT (10 MIN)

7) [Algorithm Assessment](#)

EXTENDED LEARNING

Use these activities to enhance student learning. They can be used as outside of class activities or other enrichment.

At Your Word

- Make up simple Tangram Algorithms for your class, and have them figure out how many different images they can create that follow that algorithm to the letter.
  - Choose a couple of drawings to analyze against the algorithm.

- Move toward more specific algorithms that leave little room for variation.

Shapely Debugging

- Create an algorithm for an image, and provide the class with a Tangram Arrangement that doesn't *quite* match.
  - Ask the class if the image matches the algorithm.
  - Can they figure out where it went wrong?
  - Do you need to throw the whole arrangement out and start over or can you just start from where the algorithm went wrong?

LESSON TIP *Try to focus on misplacements that allow the class to back up only a few steps to fix the algorithm. We want to get it in the students' heads that they don't have to delete entire programs if something doesn't work, they just need to find the error and fix that bit (and any bit that was relying on that instruction).*