

Instructor notes:

This is an initial activity for the start of algorithms.

Process objectives are:

The activity fosters communication skills by requesting the manager to discuss the answers with the team.

Each person in the group has a particular role to play. This fosters some dependence among the team members.

Teamwork...While the process of calculating an average should be simple for the students (the initial activity), the thoughts of how to describe that process in a generic algorithm may be. The group will work together to try to figure out how to build this algorithm and post the results on the board. This activity also introduces the students to POGIL and how it will work in the context of this classroom.

Introducing the students to the activity.

Tell the students that it is most important to have all members of the team agree on the final answers. Each member of the team should record their own answers and then compare those answers to the group responses.

Turn in the individual papers with the group response. I expect that the individual papers will be filled with cross outs as you arrive at common answers. The individual papers are your class notes for the day. NOTE: I do not require that the entire assignment be done. If most of the class needs more time, I will incorporate any missed questions into the next day's activity. It is more important to understand what you are doing than to have a completely filled out worksheet.

I will grade these based on the "valiant attempt". In other words, if I see that your individual paper is filled in similarly to the group even if there are errors, you will receive full credit for the work that day. The group paper will be assessed for accuracy, but will not be part of your grade. Be sure to review the graded worksheet with your individual worksheets to make sure that you understand the questions.

CS 139 In Class Activity – What is an Algorithm?

Content Objectives: At the conclusion of this activity students will be able to:

- Define algorithm in their own words.
- Given a familiar task, think through the steps and develop a general algorithm to accomplish that task.
- Given an algorithm, develop test cases to test the validity of the algorithm.

Introduction to POGIL Activities

- **POGIL** – Process Oriented Guided Inquiry Learning is a Learner-Centered teaching technique in which co-dependent groups work together to learn new material. We will often use these kinds of activities in lecture.
- Each person in the group has a role to play. You may choose anyone to serve in these roles, but only one role per person. The roles are:
- **MANAGER:** The **MANAGER** keeps the group on task and will be the person to ask questions of the instructor or another group if necessary. The manager also gets necessary supplies for the group and turns the folder back in at the end of each class period with the group answer sheets and other materials as specified in the activity.
- **RECORDER:** Responsible for placing group answers to **BOARD** questions up on the board. They may get help from other team members in doing so.
- **PRESENTER:** If called upon, this person will report on the group answer (that the recorder has recorded) and will put all **BOARD** questions up on the whiteboard.
- **REFLECTOR:** If there are 4 people or more in the group, reflectors think about the activity and fill out the exit pass at the end of the class session. They should indicate anything in the activity that the group had trouble with (Activity fuzzy), anything in the material content that the group still is having trouble with (Content fuzzy), and one thing that went well this day (Hooray).

In POGIL activities, group members will work together to come to some common understanding of the content material. While you may ask the instructor for help, in most cases the activity is designed to let you wrestle with some new content. For this class, most of the POGIL activities will be application problems that you will need to work through based on a pre-class tutorial that you will view prior to that day's class.

You will see **BOARD** at the start of some of the activities or questions. When you encounter a **BOARD** question, the recorder should record the group name and answer to that question(s) on the white board.

Today's activity

PART1 – What is an algorithm? In this section and for most of the activities this semester, each person should work through the question on their own, then compare answers with the group. The **MANAGER** should facilitate this work. Stop at each question to make sure that each person in the group understands the answer and to make sure that the recorder has the correct group answer on his/her sheet.

1. **WITHOUT USING A CALCULATOR**, Calculate the average (to 1 decimal place) of the following numbers. Think about the process.

5, 27, -5 _____

2. **WITHOUT USING A CALCULATOR**, calculate the average (to 1 decimal place) of the following numbers.

12, 37, 19.5, 27.5 _____

3. **WITHOUT USING A CALCULATOR**, calculate the average (to 1 decimal place) of the following numbers.

_____ (intentionally no numbers...how do you work this out?)

4. Is there any limit to the length of the series of numbers for which you can calculate an average? Why or why not?
5. Each person in the group should create another average example problem using a different series of numbers. Compare results. Which examples would you use to test a program that averages numbers?
6. **BOARD** If you were to write instructions for someone about how to calculate an average of numbers by hand, what would those instructions be? Work with this in your group. Use the examples that you created to test your algorithm. When you believe that your algorithm works well, put it on the board.
7. As other students put their answers on the board, compare their answers to yours. Think about the reasons why your instructions may seem better to you than another team's or why their instructions are better than yours.
8. These instructions are also called algorithms. In your own words, what is an algorithm?



PART 2 – Let's explore an algorithm represented in Python.

(NOTE: This example came from <https://docs.python.org/2/tutorial/controlflow.html>)

```
def fib(n):    # write Fibonacci series up to n
    """Print a Fibonacci series up to n."""
    a, b = 0, 1
    while a < n:
        print a,
        a, b = b, a+b

>>> # Now call the function we just defined:
fib(10)
```

This is a dialog of a session in IDYL (editor for Python).

1. What primitives that we have discussed are used in this example?
2. What do you see that we have not yet discussed?
3. What would be output by the call to this function (fib(10))? NOTE: The `print` statement will output the value which follows it.
4. What other examples would you use to test this? Come up with three different examples? Discuss with your team why you chose those examples and what you would expect the output to be.
5. Notice that this algorithm would print something only if $n > 0$. How would you change this algorithm to check if n is 0 or less and print an error message if so? If $n > 0$, it should process the numbers as before?

BOARD: Write your new fib function here:

6. At the conclusion of the lesson, have your reflector fill in the exit pass with input from the team. If you have no reflector, have the recorder fill that role.
-

7. References for this lesson

Brookshear, Chapter 5

POGIL.org – For more information including research findings and getting started with POGIL.

CSPOGIL.org – For more examples of POGIL (including a number from JMU) in various CS courses.

python.org – For more information about Python

<http://www.codecademy.com/tracks/python> - For learning Python