Name _____ Period _____

Solving Problems

Your Tasks (Mark these off as you go)

- ☐ Complete the problems
- ☐ Pair up and reflect
- ☐ Compare algorithms
- ☐ Wrap up
- ☐ Share out
- ☐ Receive credit for this lab guide

☐ Complete the prompts

Today we are going to explore how computer scientists think about problem-solving. An important skill we will explore will be recognizing patterns and similarities.

Review the problems below for one minute, and then move around the room and collect the information needed to solve the problems.

	Prompt	Information
1	Find a person whose birthday is before yours	
2	Find a person whose birthday is after yours	
3	Find the person whose birthday is the closest before yours	
4	Find the person whose birthday is the closest after yours	
5	Find the person whose birthday is closest to yours	
6	Find the person with an equal number of birthdays before and after theirs	
7	Find the two people with the closest birthdays in the room	
8	Find the shortest period of time in which three people have birthdays	
9	Find the shortest period of time in which four people have birthdays	
10	Find the longest period of time in which no one has a birthday	

☐ Pair up and reflect

Locate the person(s) you found for prompt 5 above – it's ok if there is more than one. Write their name(s) below. Indicate one fun fact about each person.

Person(s)	
Fun fact(s)	

Discuss the following prompts with your group. Write your responses below.
How did you go about solving each of the problems?
For which problems did you need to do something similar to solve them? For examples problems 1 & 2 are very similar.
How many people are in the room? What is the probability that two people will have the same birthday based on the number of people. How did you arrive at this number?
Watch the following video, https://youtu.be/ofTb57aZHZs . How does your prediction compare?

☐ Compare algorithms

We just thought about whether problems are similar. Now we're going to look at whether we're solving the same problem.

Consider the algorithms below. Use the markers you have been provided to trace the algorithms on your desk.

With your partner decide which of these programs are "the same" as one another.

Algorithm 1

```
MOVE_FORWARD()
TURN_RIGHT()
MOVE_FORWARD()
TURN_RIGHT()
MOVE_FORWARD()
TURN_RIGHT()
MOVE_FORWARD()
TURN_RIGHT()
```

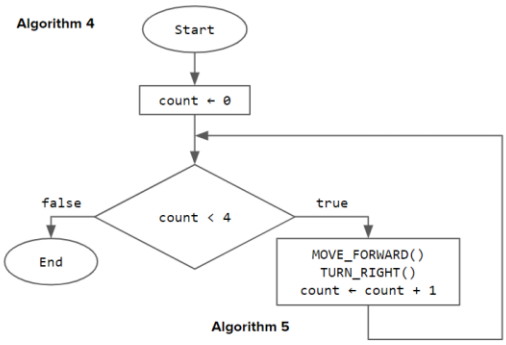
Algorithm 2

```
REPEAT 2 TIMES
{
    MOVE_FORWARD()
    MOVE_FORWARD()
    TURN_RIGHT()
    MOVE_FORWARD()
    TURN_RIGHT()
}
```

Algorithm 3

```
moves ← ["F", "R", "F", "R", "F", "R", "F", "R"]
FOR EACH move IN moves
{
    IF (move = "F")
    {
        MOVE_FORWARD()
    }
    ELSE
    {
        TURN_RIGHT()
    }
}
```

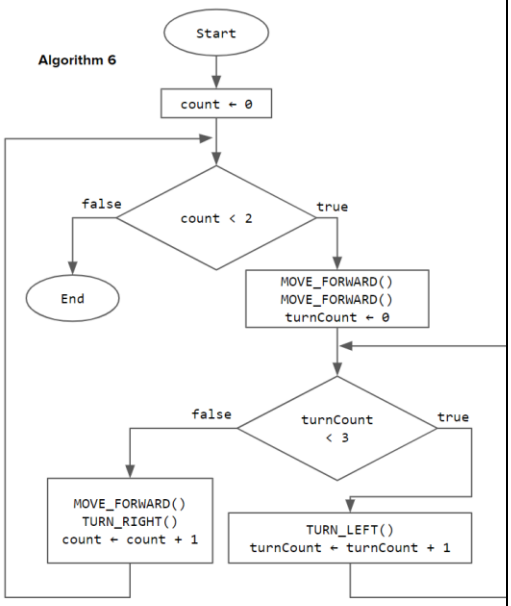
Algorithm 4



Algorithm 5

```
REPEAT 2 TIMES
{
    REPEAT 2 TIMES
    {
        MOVE_FORWARD()
    }
    REPEAT 3 TIMES
    {
        TURN_LEFT()
    }
    MOVE_FORWARD()
    REPEAT 3 TIMES
    {
        TURN_LEFT()
    }
}
```

Algorithm 6



Discuss with your group which of these algorithms are like one another? How did you decide that?

☐ Wrap-up

In the first part of this activity, you had to solve a problem. In the second part of this activity, you explored algorithms. In the space below write a definition for each of these terms.	
Problem	
Algorithm	

☐ Share out

Pick a spokesperson for your group. When it's their turn share the following with the class,

- Your name and a fun fact
- Your partner's name (or names if you have more than one) along with a fun fact about each

☐ Receive credit for this lab guide

Submit this portion of the lab to Pluska to receive credit for the lab guide.