

The Soggy Biscuits  
Samuel Mitchell :3.3.1  
Andrew Martin :3.4.1  
Charles Ceccardi: 3.2.1  
Mitchell Oneka: 3.1.1

### 3.1.1 [10 min] Critical Thinking Questions

1. What are the recursive steps and base cases for Fibonacci when  $n < 0$ ?

The recursive step for negative numbers in the Fibonacci sequence is  $(-1)^{n+1} * f(-n)$ . The base case is where  $f(0) < 0$ .

2. What is missing from the above Fibonacci code (Base Case and Recursive Step)?

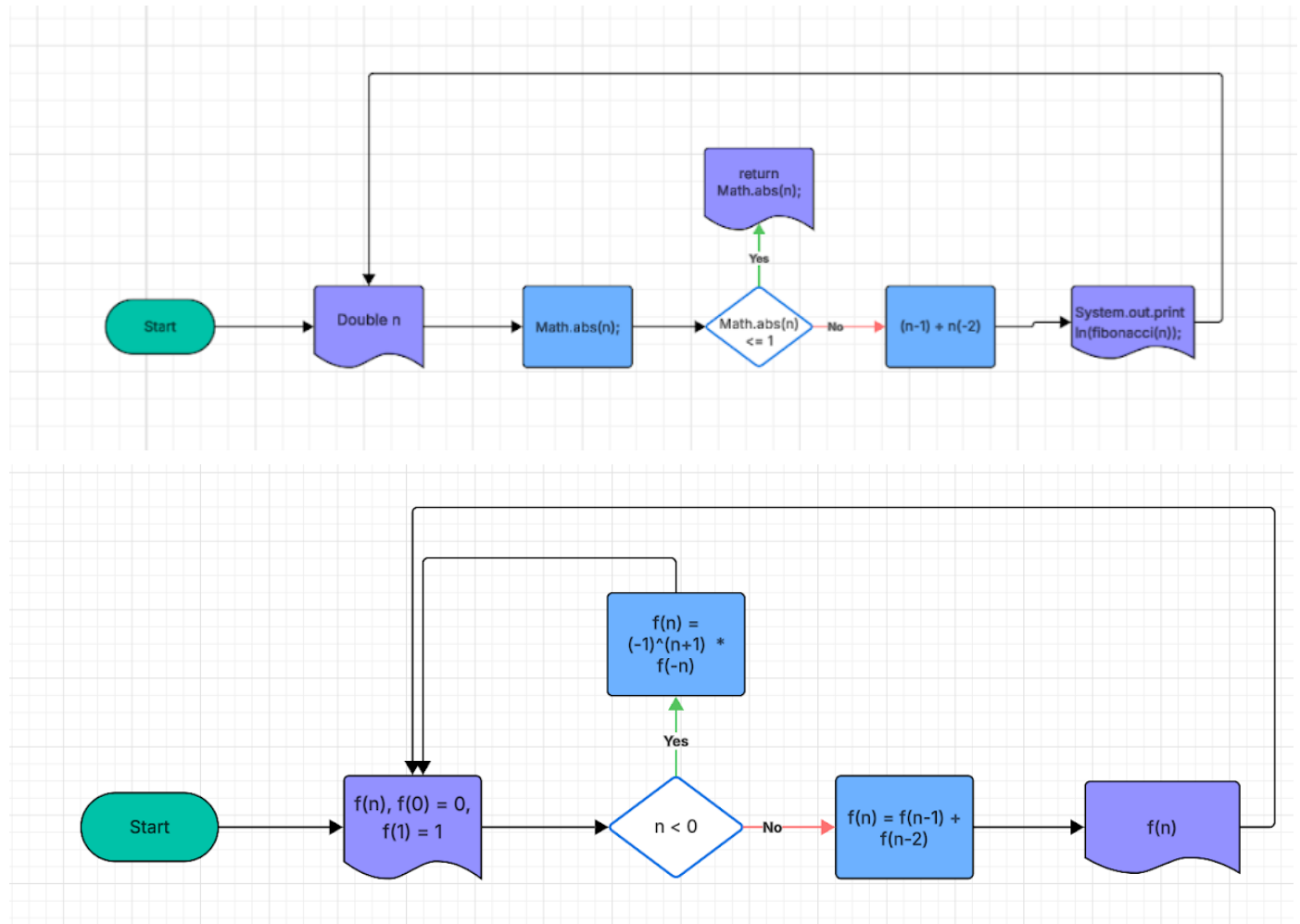
Implementing negative numbers:

```
if (n < 0) {  
    return Math.pow(-1, (n + 1)) * f(-n);  
}
```

Test code:

```
public static void main (String args[]) {  
    int n = 2;  
    System.out.println(fibonacci(n));  
}
```

3. Make a flow chart for the Fibonacci method.



### 3.2.1 Critical Thinking Questions

1. Develop pseudocode for two different solutions to this problem. (You only need to implement one solution.)

//1.

//if  $n$  is  $< 0$ . Recall sumDigit with  $n$  as positive. Then make the value negative again and return.

//convert  $n$  to a string, call string digit

//make a int for sum

//search digit and add up sum

//return sum

//2.

//if  $n$  is  $< 0$ . Recall sumDigit with  $n$  as positive. Then make the value negative again and return.

//convert  $n$  to a string, call string digit

//take last digit to add to sum

//make a int for a number in  $n$

//shorten  $n$  for next call

//end recursion when n = 0

2. What are the recursive steps and base cases for your implementation?

The base case is n, n gets smaller in each call and when n becomes 0 recursion ends. Subtract and divide from n each call for the next call until n becomes 0.

### 3.3.1 Critical Thinking Questions

1. How can you test this method?

By throwing weird edge cases to find if it still says true

2. What are the unusual cases (also called edge cases) that you have to test?

Removing spaces and making sure to make the string lowercase when doing it as to not have any discrepancies.

### 3.4.1 Critical Thinking Questions

1. Suppose the input array could be null or empty. What would reasonable results be?

Reasonable results could be zero, null, or a string reading "invalid input"

2. How would you test for these results?

Use a helper method to filter edge cases before calling the real method.

**Process Analyst:** Consider the group's performance today. Email one piece of positive evidence to each member of the group for one of the Effectiveness Competencies listed below. Only give each person one piece of positive

feedback right now. Interpersonal Skills Speaks clearly, confidently, and with appropriate eye-contact with audience. Applies active listening skills using reflection, restatement, questioning, and clarification.

Integrity Treats others honestly, fairly, and courteously. Accepts responsibility for one's decisions and actions.

Professionalism Accepts feedback and attempts to learn from mistakes or misunderstandings. Demonstrates positive attitude toward work and others. Manages conflict respectfully while maintaining progress toward a goal.

Lifelong Learning seeks feedback and asks questions to strengthen own understanding. Uses newly learned knowledge and skills to complete specific tasks.

**Quality Control:** Summarize the lab activities in the Week10LabReport document by summarizing each part and explaining the group's solutions.

For the palindrome checker I traversed the string backwards and added the chars to a string to let it be reversed.

**Facilitator:** Engage all members of the group in a discussion of the questions below. Submit the group's answers to Canvas with the Week10Lab.zip file.

1. Is Test-Driven Development (TDD) a test practice or a design practice?

It is a design practice.

2. Is it impossible to generate a good design without doing TDD?

Yes, however TDD makes the creation of a good design much easier.

3. How do tests help you create a better design, i.e. classes with low coupling, high cohesion, simple code, etc.?

TDD ensures that programmers can focus on making the code do exactly what is necessary, no more and no less.

4. Now that I've used my tests to improve the design of the system, can I throw them away?

No, the tests are useful to keep for future iterations of the project.

5. Some people say that writing tests before programming decreases productivity (for example, if a programmer writes a hundred lines of code a day, he is going to write 50 lines of tests and 50 lines of production code). What do you think about it?

It makes sense in the context of something small like this lab, however when it comes to larger projects, it is much more useful to use TDD because it guides the programmer towards the exact problem they should be solving.

**Spokesperson:** Gather together all of the work from the lab and prepare it for submission. Combine everything into a zip file called Week10Lab.zip. Then make sure all team members have a copy to submit. Finally, present the group's work to the Lab Instructor for grading.

linux:

```
java -cp ../lib/junit-4.13.2.jar:../lib/hamcrest-core-1.3.jar:src/ org.junit.runner.JUnitCore
Week10LabTest
```

```
javac -cp ../lib/junit-4.13.2.jar:../lib/hamcrest-core-1.3.jar src/*.java
```

Windows:

```
javac -cp "lib/junit-4.13.2.jar;lib/hamcrest-core-core.jar;src" src/Week10Lab.java
src/Week10LabTest.java
```