Whiteboard grading rubric for:
Date: Start time: End Time:

1. Interpreted the question
a. ___/2 points: Visually illustrated the problem domain
b. ___/2 points: Identified inputs and outputs
c. ___/2 points: Identified correct data structure

2. Solved the technical problem
a. ___/4 points: Solution works
b. ___/4 points: Code was syntactically correct
c. ___/4 points: Code was idiomatically correct
d. ___/4 points: Solution was the best possible option

3. Analyzed the proposed solution
a. ___/2 points: Stepped through their solution
b. ___/2 points: Big O time and space are considered

4. Communicated effectively throughout

a. ___/4 points:Verbalized their thought process
b. ___/2 points:Used correct terminology
c. ___/2 points:Used the time available effectively
d. ___/2Was not overconfident (not listening to suggestions)

e. ___/2 points:Was not underconfident (unsure of known algorithm)
f. ___/2 points:Whiteboard was readable (penmanship and spacing)

___/40 Total points

Giving up is an automatic fail, 80% required to pass

Problem 1

Given a calculator constructor

```
function calculator(){
   this.result = null;
}
```

1. Add a function to provide ‘subtract’ that takes a single input and subtracts from the result
2. Add a function to provide ‘divide’ that takes a single input and divides the result
3. Clear function to clear result.
4. Stretch make the functions chainable
5. HARD: Add a function to provide ‘fibonacci’ that returns the final value in a fibonacci series starting at 0 and ending at the value of result: for example if result is 5 then fib(5) = [0, 1, 1, 2, 3, 5] = 5. If result is 0 then fib(0) = 0, fib(1) = 0, fib(2) = 0, fib(3) = 2, fib(4) = 3. Instruct the whiteboarder to calculate the fibonacci series for the value of this result and return the value at length - 1 of the series array.

Look for edge cases:
Prevent divide by 0 if applicable
Allow for non integer results calculations - you can tell than about the parseFloat function
Don’t try to calculate a fibonacci series on a non-integer result value
Encourage the whiteboarder to ask questions and provide answers based on your knowlege of solutions.

```
calculator.prototype.subtract = function(val) {
  if (this.result === null) this.result = val;
  else(parseFloat(this.result -= val));
  return this; // this gives chaining
};
//what happens if val is 0
calculator.prototype.divide = function(val) {
  If (val === 0) {
    console.log(‘no divide by zero’);
    return;
  }
  if (this.result === null) this.result = val;
  else(parseFloat(this.result /= val));
  return this; // this gives chaining
};
calculator.prototype.clear = function() {
  this.result = null;
  return this;
};
```

Chained Calc:

```
var calc = new Calculator();
calc.subtract(3).divide(2);
function fibSeries(n) {
  var f = [];
  for (var c = 0; c <= n; ++c) {
    f.push((c < 2) ? c : f[c - 1] + f[c - 2]);
  }
  return f;
}
calculator.prototype.fibonacci = function() {
  if (!Number.isInteger(this.result)) {
    console.log('must be an integer for this function');
    return;
  }
  var series = fibSeries(this.result);
  console.log(series);
  return series[series.length - 1];
};
```