

# STEVE

MICHAEL O'CAIN

CHRIS TUCKER

CONNOR YATES

## INTRODUCTION

## METHODOLOGY

## RESULTS

## WHAT WE TESTED

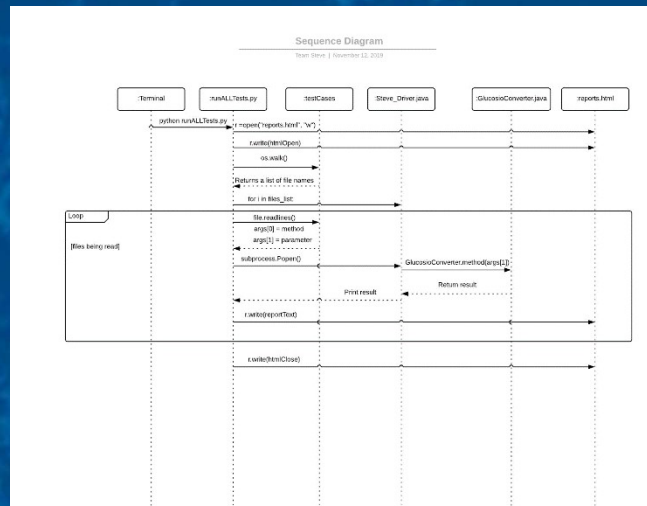
Team Steve tested Glucosio, an HFOSS app designed to help its users monitor their glucose level. We developed a driver to test five methods used in the Glucosio tool package: `round()`, `glucoseToMgDL()`, `glucoseToMmolL()`, `glucoseToA1C()`, `a1cToGlucose()`. All of these methods perform rounding operations to calculate various double values.

## OUR DRIVER

```
21 public static void main(String args[]){
22
23     String methodName = args[0];
24     double argument = Double.parseDouble(args[1]);
25     //double out = -1.0;
26     Object out;
27     int argument2;
28     switch (methodName) {
29
30         default:
31             //argument2=Integer.parseInt(args[2]);
32             //out = GlucosioConverter.round(argument,2);
33             out= "Invalid method called";
34             break;
35
36         case "round":
37             //argument2=Integer.parseInt(args[2]);
38             out = GlucosioConverter.round(argument,2);
39             //out = GlucosioConverterOriginal.round(argument,2);
40             break;
41
42         case "glucoseToMgDL":
43
44             out = GlucosioConverter.glucoseToMgDL(argument);
45             break;
46
47         case "glucoseToMmolL":
48
49             out = GlucosioConverter.glucoseToMmolL(argument);
50             //out = GlucosioConverterOriginal.glucoseToMmolL(argument);
51             break;
52
53         case "glucoseToA1C":
54
55             out = GlucosioConverter.glucoseToA1C(argument);
56             //out = GlucosioConverterOriginal.glucoseToA1C(argument);
57             break;
58
59         case "a1cToGlucose":
60
61             out = GlucosioConverter.a1cToGlucose(argument);
62             //out = GlucosioConverterOriginal.a1cToGlucose(argument);
63             break;
64
65     }
66
67     System.out.println(out);
68 }
```

## OUR TESTING FRAMEWORK

We developed a script that would read test cases from a folder and give their contents to our driver, which would in turn call the methods being tested in the Glucosio package. The script would then capture the results from the driver and record them in an html file.



## RESULTS FROM TESTING

All of the methods we tested depended on the round method and passed regardless of what integer value was provided. Below is an example output.

GlucosioConverter.java Test									
File	Test	Method	Test Description	Parameter	Result	Output	Result	Pass	Fail
Test Case 1	round	round	Round integer i double and return it to the nearest hundredth place. If the double has two trailing zeros the rounded integer is double with only one trailing zero.	1.2345	1.23	1.23	1.23	Pass	Fail
Test Case 2	round	round	Round integer i double and return it to the nearest hundredth place. If the double has two trailing zeros the rounded integer is double with only one trailing zero.	1.2345	1.23	1.23	1.23	Pass	Fail
Test Case 3	round	round	Round integer i double and return it to the nearest hundredth place. If the double has two trailing zeros the rounded integer is double with only one trailing zero.	1.2345	1.23	1.23	1.23	Pass	Fail
Test Case 4	round	round	Round integer i double and return it to the nearest hundredth place. If the double has two trailing zeros the rounded integer is double with only one trailing zero.	1.2345	1.23	1.23	1.23	Pass	Fail
Test Case 5	round	round	Round integer i double and return it to the nearest hundredth place. If the double has two trailing zeros the rounded integer is double with only one trailing zero.	1.2345	1.23	1.23	1.23	Pass	Fail
Test Case 6	glucoseToMgDL	glucoseToMgDL	glucoseToMgDL converts the input glucose to Mg/dL	1.0	18.0	18.0	18.0	Pass	Fail
Test Case 7	glucoseToMgDL	glucoseToMgDL	glucoseToMgDL converts the input glucose to Mg/dL	1.0	18.0	18.0	18.0	Pass	Fail
Test Case 8	glucoseToMgDL	glucoseToMgDL	glucoseToMgDL converts the input glucose to Mg/dL	1.0	18.0	18.0	18.0	Pass	Fail
Test Case 9	glucoseToMgDL	glucoseToMgDL	glucoseToMgDL converts the input glucose to Mg/dL	1.0	18.0	18.0	18.0	Pass	Fail
Test Case 10	glucoseToMgDL	glucoseToMgDL	glucoseToMgDL converts the input glucose to Mg/dL	1.0	18.0	18.0	18.0	Pass	Fail
Test Case 11	glucoseToMgDL	glucoseToMgDL	glucoseToMgDL converts the input glucose to Mg/dL	1.0	18.0	18.0	18.0	Pass	Fail
Test Case 12	glucoseToMgDL	glucoseToMgDL	glucoseToMgDL converts the input glucose to Mg/dL	1.0	18.0	18.0	18.0	Pass	Fail
Test Case 13	glucoseToMgDL	glucoseToMgDL	glucoseToMgDL converts the input glucose to Mg/dL	1.0	18.0	18.0	18.0	Pass	Fail
Test Case 14	glucoseToMgDL	glucoseToMgDL	glucoseToMgDL converts the input glucose to Mg/dL	1.0	18.0	18.0	18.0	Pass	Fail
Test Case 15	glucoseToMgDL	glucoseToMgDL	glucoseToMgDL converts the input glucose to Mg/dL	1.0	18.0	18.0	18.0	Pass	Fail

We injected faults into the Glucosio code to exercise our framework and find any faults that might otherwise not have been detected. Below is an example output.

GlucosioConverter.java Test									
File	Test	Method	Test Description	Parameter	Result	Output	Result	Pass	Fail
Test Case 1	round	round	Round integer i double and return it to the nearest hundredth place. If the double has two trailing zeros the rounded integer is double with only one trailing zero.	1.2345	1.23	1.23	1.23	Pass	Fail
Test Case 2	round	round	Round integer i double and return it to the nearest hundredth place. If the double has two trailing zeros the rounded integer is double with only one trailing zero.	1.2345	1.23	1.23	1.23	Pass	Fail
Test Case 3	round	round	Round integer i double and return it to the nearest hundredth place. If the double has two trailing zeros the rounded integer is double with only one trailing zero.	1.2345	1.23	1.23	1.23	Pass	Fail
Test Case 4	round	round	Round integer i double and return it to the nearest hundredth place. If the double has two trailing zeros the rounded integer is double with only one trailing zero.	1.2345	1.23	1.23	1.23	Pass	Fail
Test Case 5	round	round	Round integer i double and return it to the nearest hundredth place. If the double has two trailing zeros the rounded integer is double with only one trailing zero.	1.2345	1.23	1.23	1.23	Pass	Fail
Test Case 6	glucoseToMgDL	glucoseToMgDL	glucoseToMgDL converts the input glucose to Mg/dL	1.0	18.0	18.0	18.0	Pass	Fail
Test Case 7	glucoseToMgDL	glucoseToMgDL	glucoseToMgDL converts the input glucose to Mg/dL	1.0	18.0	18.0	18.0	Pass	Fail
Test Case 8	glucoseToMgDL	glucoseToMgDL	glucoseToMgDL converts the input glucose to Mg/dL	1.0	18.0	18.0	18.0	Pass	Fail
Test Case 9	glucoseToMgDL	glucoseToMgDL	glucoseToMgDL converts the input glucose to Mg/dL	1.0	18.0	18.0	18.0	Pass	Fail
Test Case 10	glucoseToMgDL	glucoseToMgDL	glucoseToMgDL converts the input glucose to Mg/dL	1.0	18.0	18.0	18.0	Pass	Fail
Test Case 11	glucoseToMgDL	glucoseToMgDL	glucoseToMgDL converts the input glucose to Mg/dL	1.0	18.0	18.0	18.0	Pass	Fail
Test Case 12	glucoseToMgDL	glucoseToMgDL	glucoseToMgDL converts the input glucose to Mg/dL	1.0	18.0	18.0	18.0	Pass	Fail
Test Case 13	glucoseToMgDL	glucoseToMgDL	glucoseToMgDL converts the input glucose to Mg/dL	1.0	18.0	18.0	18.0	Pass	Fail
Test Case 14	glucoseToMgDL	glucoseToMgDL	glucoseToMgDL converts the input glucose to Mg/dL	1.0	18.0	18.0	18.0	Pass	Fail
Test Case 15	glucoseToMgDL	glucoseToMgDL	glucoseToMgDL converts the input glucose to Mg/dL	1.0	18.0	18.0	18.0	Pass	Fail

