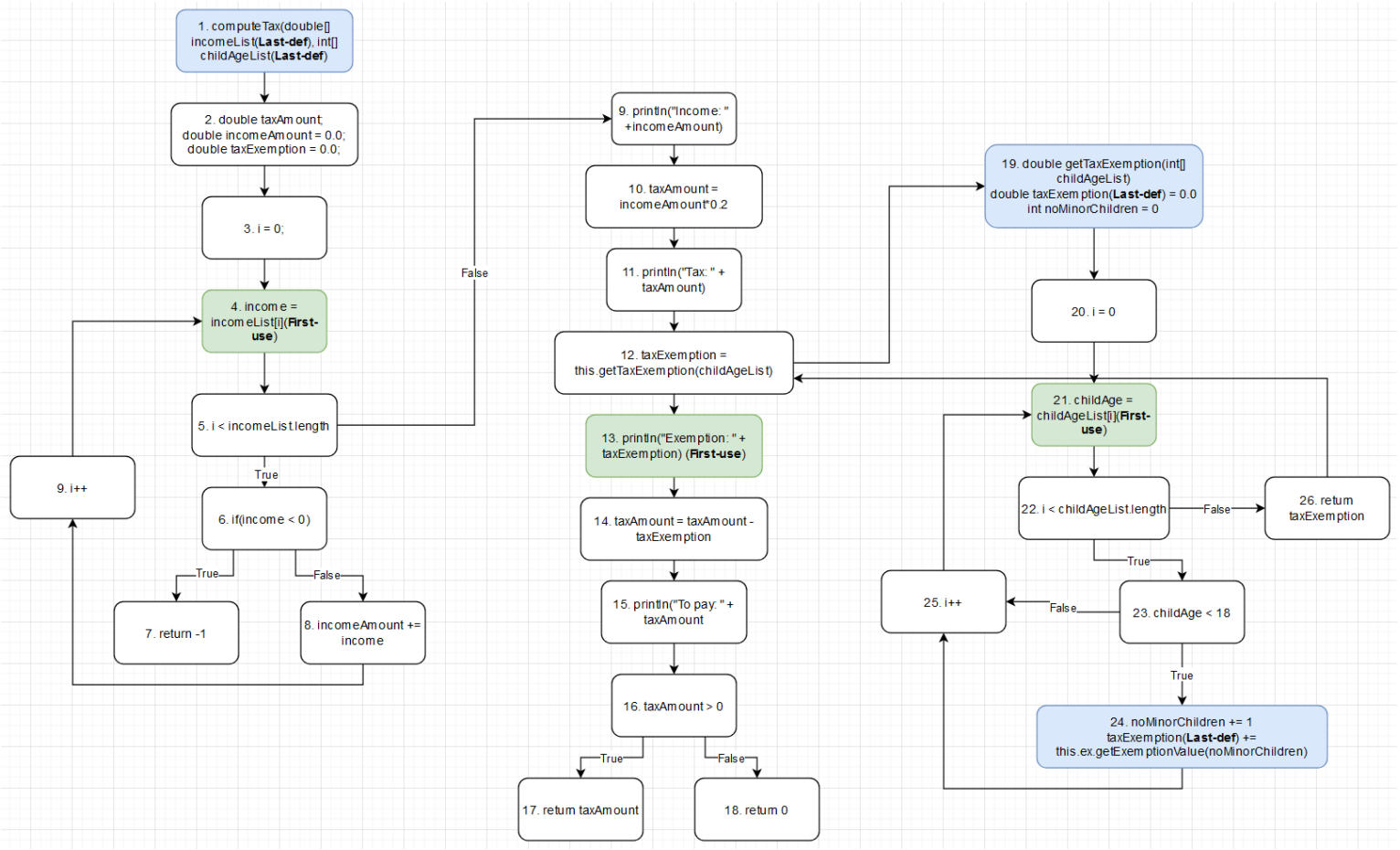
Assignment 2

Task 1

1.



Blue: **Last-def**, Green: **First-use**

2.

childAgeList:

1 -> 21

[1,2,3,4,5,9,10,11,12,19,20,21]

AND

taxExemption:

19 -> 13

[19,20,21,22,26,12,13]

OR

24 -> 13

[24,25,21,22,26,12,13]

3.

childAgeList:

1 -> 21

[1,2,3,4,5,9,10,11,12,19,20,21]

AND

taxExemption:

19 -> 13

[19,20,21,22,26,12,13]

AND

24 -> 13

[24,25,21,22,26,12,13]

4 and 5.

Test	Test Path	Input	Expected
T1	[1,2,3,4,5,9,10,11,12,19,20,21,22,26,12,13,14, 15,16,17]	childAgeList[] = [] incomeList[] = []	0
T2	[1,2,3,4,5,9,10,11,12,19,20,21,22,23,24,25,21, 22,26,12,13,14,15,16,17]	childAgeList[] = [12] incomeList[] = []	0

Task 2

I removed the mutations that had nothing to do with either calls or returns and was left with 10 mutations. The mutations removed were the mutants on like 21, 24, 28, 35, 39, 53, 54. Since a print command is also a call to java/io/PrintStream it has been included. Due to 7 of the mutations surviving the mutation covarage falls to 30% or 3/10 as the program survives without the print statements.

Mutations

```
8  1. removed call to java/io/PrintStream::println → SURVIVED
14 1. removed call to java/io/PrintStream::println → SURVIVED
21 1. negated conditional → KILLED
   2. changed conditional boundary → SURVIVED
23 1. replaced double return with 0.0d for assigs/assig2/TaxCalculator2::computeTax → NO_COVERAGE
24 1. Replaced double addition with subtraction → KILLED
26 1. removed call to java/io/PrintStream::println → SURVIVED
28 1. Replaced double multiplication with division → KILLED
30 1. removed call to java/io/PrintStream::println → SURVIVED
33 1. removed call to java/io/PrintStream::println → SURVIVED
35 1. Replaced double subtraction with addition → KILLED
36 1. removed call to java/io/PrintStream::println → SURVIVED
39 1. negated conditional → KILLED
   2. changed conditional boundary → SURVIVED
40 1. replaced double return with 0.0d for assigs/assig2/TaxCalculator2::computeTax → KILLED
53 1. negated conditional → SURVIVED
   2. changed conditional boundary → SURVIVED
54 1. Replaced integer addition with subtraction → KILLED
55 1. Replaced double addition with subtraction → KILLED
58 1. replaced double return with 0.0d for assigs/assig2/TaxCalculator2::getTaxExemption → KILLED
```

Number of Classes	Line Coverage		Mutation Coverage		Test Strength	
1	96%	27/28	45%	9/20	47%	9/19

Breakdown by Class

Name	Line Coverage		Mutation Coverage		Test Strength	
TaxCalculator2.java	96%	27/28	45%	9/20	47%	9/19

This is the original mutation score for TaxCalculator2 but with only mutators for integration testing mutation coverage falls to 3/10 and the test strength becomes 3/9 due to one of the mutations not having coverage. The final mutation score with only integration mutations is:

Line Coverage: 27/28

Mutation Coverage: 3/10

Test Strength: 3/9

The score being so low is because the print statements are not being tested for.