SQE — Assignment 3: Symbolic Execution

1. Introduction

In this assignment, you will symbolically execute a function, and find an input vector for a specific minimal set.

2. Goals

Practice symbolic execution and understand its relation to CFG.

3. The assignment task

Consider the following code and its CFG:

```
1.   public void run(int y, int z) {
2.     if (y < 6) {
3.       System.out.println("Some code");
4.     } else {
5.       System.out.println("Some more code");
6.     }
7.     while ((z == 1 && y != 4)) {
8.       System.out.println("Some while loop code");
9.       z++;
10.    }
11.  }
12.
13.  void main(String[] args) {
14.    int y = args[0];
15.    int z = args[1];
16.    run(y, z);
17.  }
```

```
strict digraph G {
  1 [ label="1: Start" shape="box" ];
  2 [ label="2: init y, z" shape="box" ];
  3 [ label="3: y<6" shape="diamond" ];
  4 [ label="4: sout(\"Some code\")" shape="box" ];
  11 [ label="11" shape="circle" ];
  6 [ label="6: sout(\"Some more code\")" shape="box" ];
  9 [ label="9: sout(\"Some while loop code\")" shape="box"];
  10 [ label="10: z++" shape="box"];
  8.1 [ label="8.1: z == 1" shape="diamond" ];
  8.2 [ label="8.2: y != 4" shape="diamond" ];
  12 [ label="12" shape="circle" ];
  13 [ label="13: End" shape="box" ];
  7 [ shpae="circle" ]
  1 -> 2 [ label="T" ];
  2 -> 3 [ label="T" ];
  3 -> 4 [ label="T" ];
  3 -> 6 [ label="F" ];
  4 -> 7 [ label="T" ];
  6 -> 7 [ label="T" ];
  7 -> 8.1 [ label="T" ];
  8.1 -> 8.2 [ label="T" ];
  8.1 -> 12 [ label="F" ];
  8.2 -> 12 [ label="F" ];
  8.2 -> 9 [ label="T" ];
  9 -> 10 [ label="T" ];
  10 -> 11 [ label="T" ];
  11 -> 8.1 [ label="T" ];
  12 -> 13 [ label="T" ];
}
```

The CFG tool (from the previous assignment) says that the following sets are minimal for branch coverage:

Set number 3:

1-2-3-6-7-8.1-8.2-12-13

1-2-3-4-7-8.1-8.2-9-10-11-8.1-12-13

Set number 6:

1-2-3-6-7-8.1-8.2-9-10-11-8.1-12-13

1-2-3-4-7-8.1-8.2-12-13

Your goal is to find input values for these minimal sets, if possible. If it is not possible — declare that the set is infeasible.

For this goal, you will analyze the code using symbolic execution and find the PCT for each path in these sets. If the PCT is satisfiable — write an input vector that satisfies it. Otherwise, explain why it is not satisfiable.

NOTE:

The code includes a while loop. While there may be infinite iterations, you may assume that there are only two iterations. You can mark these iterations in the line number. For example:

$[9_1]$ and $[9_2]$ will denote the first and the second times you arrived at line 9, respectively.

4. Submission Constraints

Your symbolic execution analysis must follow the syntax presented in the class/practical session. You can look at the solution of 2022 moed A. Your answer should use the same table format as in question 4a and a similar PCT explanation format as in 4b.

5. Deadline

The files should be submitted through Moodle by **13/02/2024 23:59**.

GOOD LUCK