**ASSIGNMENT 6**

Symbolically execute the following program, and determine whether the exception can or cannot be thrown.

Motivate your answer and show the path conditions you generated during the analysis also stating for each of them whether it is satisfiable or not.

If the exception can be thrown, provide concrete inputs that lead the system to reach it, and show that these input are a solution the path condition computed with the symbolic execution.

public class Bar {

public int el;

public Bar() {

this.el = 0;

}

public Bar(int x) {

this.el = x;

}

}

public class Test {

private Bar b1;

private Bar b2;

public int foo(int a1, int a2) throws Exception {

b1.el = a1;

b2.el = a2;

if (a1 == a2) return -1;

int z = b1.el;

z = z - b2.el;

z = 2 \* z;

if (z > 0) {

z = b1.el;

} else if (z < 0){

z = b2.el;

} else {

throw new Exception( "z cannot be zero!");

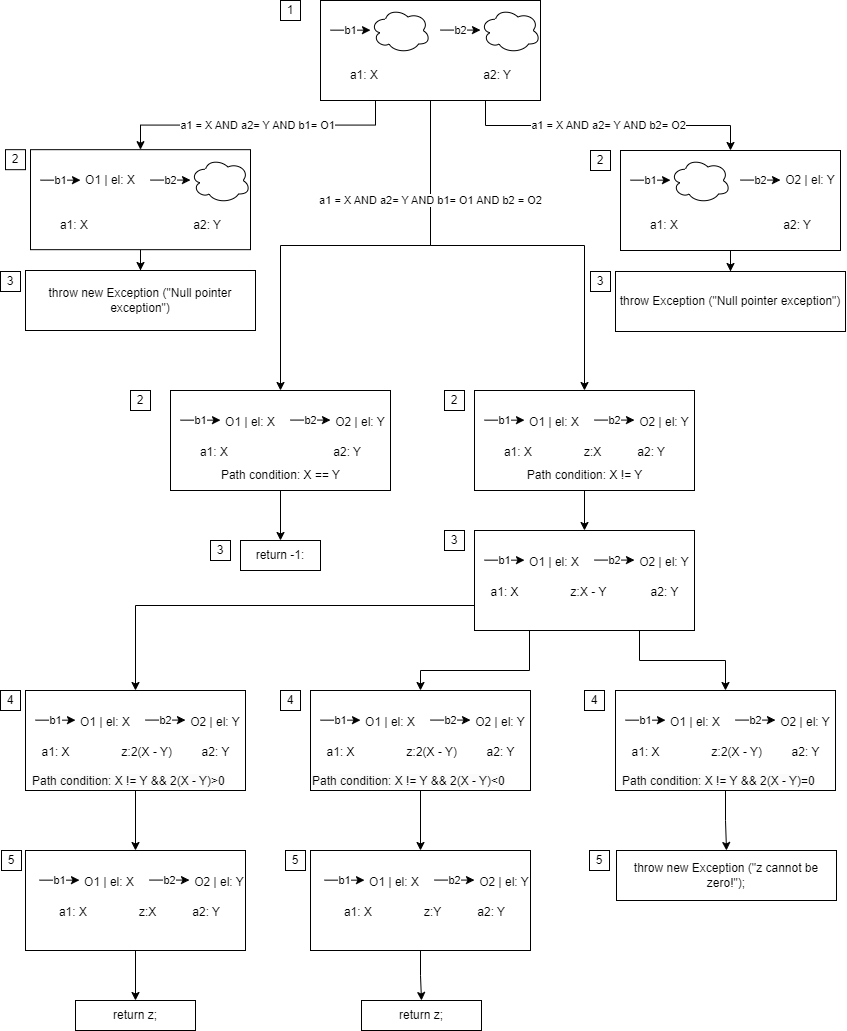
}

return z;

}

}

**Generalized Symbolic execution:**

****

Using the Symbolical execution it is possible to derive the following paths (excluding the ones that lead to a Null pointer exception):

* **Path 1**:

b1.el = a1;

b2.el = a2;

if (a1 == a2)

This path has the following path condition: a1=a2. This path is **satisfiable**, and it returns -1.

* **Path 2**:

b1.el = a1;

b2.el = a2;

int z = b1.el;

z = z - b2.el;

z = 2 \* z;

if (z > 0) {

z = b1.el;

return z;

This path has the following path condition: a1 != a2 && 2 \* (a1 - a2) > 0. This path is **satisfiable**, and it returns b1.el, which is a1.

* **Path 3**:

b1.el = a1;

b2.el = a2;

int z = b1.el;

z = z - b2.el;

z = 2 \* z;

else if (z < 0){

z = b2.el;

return z;

This path has the following path condition: a1 != a2 && 2 \* (a1 - a2) < 0. This path is **satisfiable**, and it returns b2.el, which is a2.

* **Path 4**:

b1.el = a1;

b2.el = a2;

int z = b1.el;

z = z - b2.el;

z = 2 \* z;

if (z == 0){

throw new Exception( "z cannot be zero!");

This path has the following path condition: a1 != a2 && 2 \* (a1 - a2) = 0. This path is **NOT** **satisfiable** since there arent any inputs that can satisfy the path condition.

**NON SONO PER NIENTE CONVINTO DEL GRAFICO SOPRATTUTTO QUANDO SI CREANO LE 3 STRADE IN BASE AL VALORE DI z**

**NON SO SE L’AGGIORNAMENTO DEL VALORE DI z VENGA RAPPRESENTATO COME HO FATTO IO MA NON CREDO SINCERAMENTE**

**FORSE 2(X-Y) ANDAVA MESSO GIÀ NEL NODO 3**

**NON SO SE VANNO MANTENUTE LE B1🡪 O\el**

**NON SO CHE CONSTRAINTS SCRIVERE DOPO L’INIZIO**

**NON SONO SICURO CHE I NUMERI DI FIANCO AI RIQUADRI SIA GIUSTO. HO MESSO I NUMERI CHE IDENTIFICANO UN “LIVELLO” PER FARE VEDERE COSA SUCCEDE ALLO STESSO LIVELLO IN SITUAZIONI DIVERSE, MA NON SO SE SI DEVONO USARE I NUMERI AD ESEMPIO PER IDENTIFICARE I PATH OPPURE PER INDICARE LA LINEA DI CODICE A CUI FA RIFERIMENTO IL RIQUADRO**