Assignment 10

Exercise 1

Recall (see slide 19 from the lecture “Applications”) that an interval transformer for an *action* has the type:

$$\llbracket action \rrbracket_i : (\mathit{Var} \mapsto L^i) \mapsto (\mathit{Var} \mapsto L^i)$$

where L^i are the elements of the interval domain ($L^i = \{[x, y] \mid x, y \in \mathbb{Z}^\infty, x \leq y\} \cup \{\bot_i\}$).

1. Consider the interval maps:

   $m_1 = x \mapsto [-3, 8], y \mapsto [0, 5]$

   $m_2 = x \mapsto [-3, 8], y \mapsto \bot_i$

   The interval transformer for $\leq$ is defined on slide 29. Apply the transformer to compute the result of:

   $\llbracket x \leq y \rrbracket(m_1) =$ $\qquad$ $\llbracket x \leq y \rrbracket(m_2) =$

   $\llbracket 3 \leq 5 \rrbracket(m_1) =$ $\qquad$ $\llbracket 3 \leq 5 \rrbracket(m_2) =$

   $\llbracket 5 \leq 3 \rrbracket(m_1) =$ $\qquad$ $\llbracket 5 \leq 3 \rrbracket(m_2) =$

2. Define the interval transformer for assignment:

   $\llbracket x := a \rrbracket_i(m) =$

3. Define the multiplication expression for interval elements:

   $\langle a_1 * a_2, m \rangle \Downarrow_i$?

4. Define the interval transformer for equality test:

   $\llbracket x = y \rrbracket_i(m) =$

Exercise 2

Consider the following program:

```
  foo (int x) {
1:      y := 2
2:      if (x <= y)
3:          z := 3 * x
        else
4:          z := y
5:      z := y * z
6: }
```

1. Give two concrete traces t_1 and t_2 of the program.
2. Apply the interval abstraction function α^i (similarly to slide 15) on the set $\{t_1, t_2\}$.
3. Compute the least fixpoint $\mathsf{lfp}F^i$ of the program using the interval domain abstraction. For this exercise, you can consider that the entry/initialization transformer sets all the variables (function arguments and local variables) to Top.
4. Give a concrete trace $t \in \gamma^i(\mathsf{lfp}F^i)$ that is not a valid trace. Here γ^i is the concretization function.

Exercise 3

Give two programs that are output equivalent (i.e. for the same initial state they result in the same final state) under the concrete domain, and they are not output equivalent under the interval domain.