



Tutorial for Program Verification

Exercise Sheet 10

Exercise 1: Execution of trace abstraction

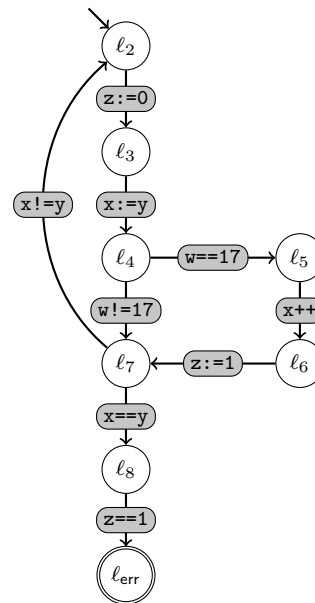
2 Points

Consider the following program and the corresponding control automaton \mathcal{A}_P .

```

int x, y, z, w;
void foo()
{
1:   do {
2:       z = 0;
3:       x = y;
4:       if (w == 17){
5:           x++;
6:           z = 1;
       }
7:   } while(x!=y)
8:   assert (z != 1);
}

```



Give two error traces π_1, π_2 and construct corresponding interpolant automata $\mathcal{A}_1, \mathcal{A}_2$ such that the inclusion $\mathcal{L}(\mathcal{A}_P) \subseteq \mathcal{L}(\mathcal{A}_1) \cup \mathcal{L}(\mathcal{A}_2)$ holds.

Remark: We call a trace π infeasible if $post(\text{true}, \pi) = \text{false}$ holds.

Exercise 2: Interpolant automata

1 Point

Prove that an interpolant automaton accepts only infeasible traces.