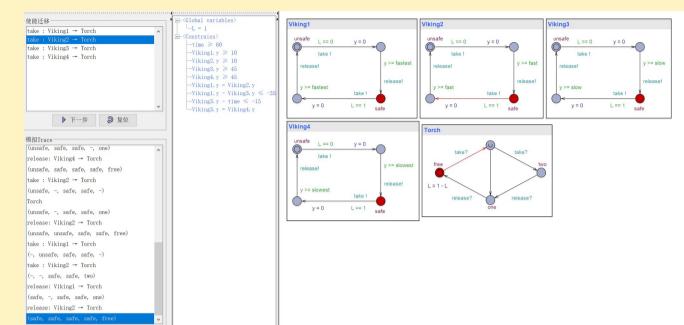
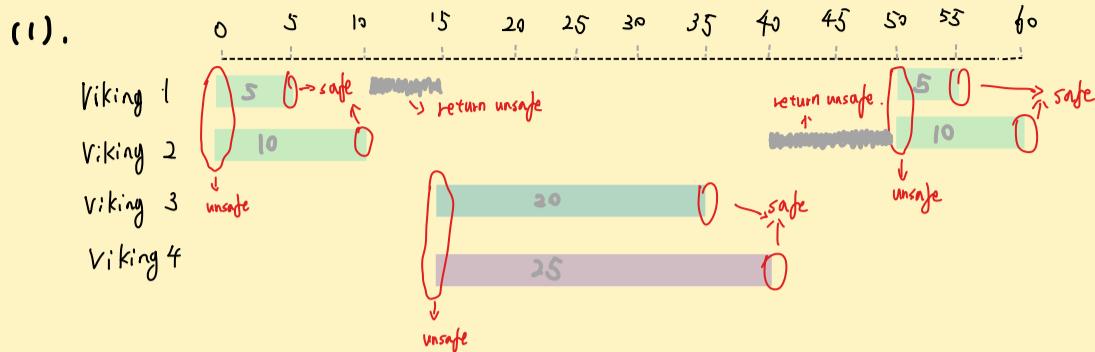
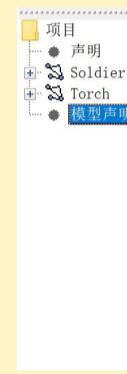
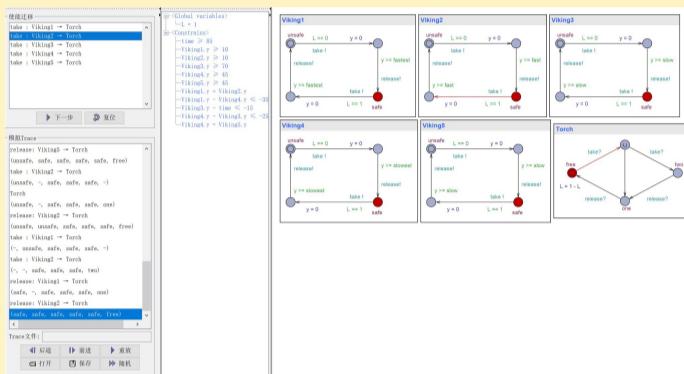


Part A.



when time = 50 min. which is possible all four Vikings get across the bridge.

(2).



```
const int fastest = 5;
const int fast = 10;
const int slow = 20;
const int slowest = 25;

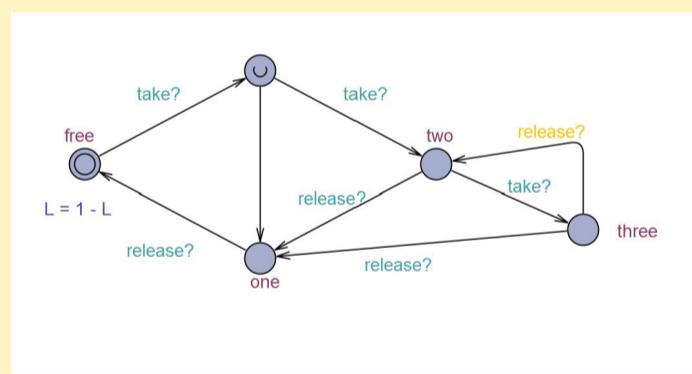
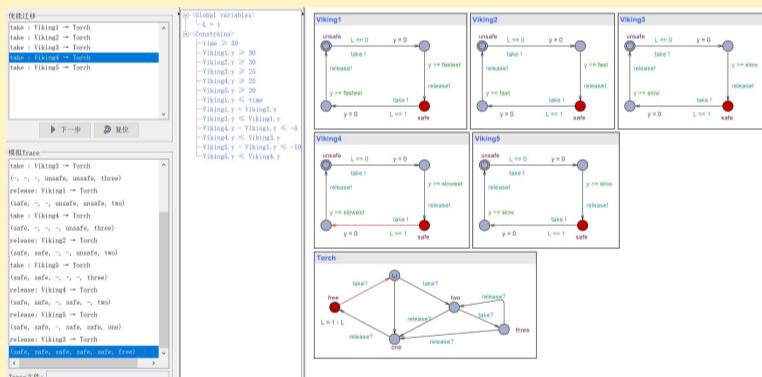
Viking1 = Soldier(fastest);
Viking2 = Soldier(fast);
Viking3 = Soldier(slow);
Viking4 = Soldier(slowest);
Viking5 = Soldier(slow);

system Viking1, Viking2, Viking3, Viking4, Viking5, Torch;
```

E<> Viking1.safe and Viking2.safe and Viking3.safe and Viking4.safe and Viking5.safe

shortest time : 85 mins.

(3)



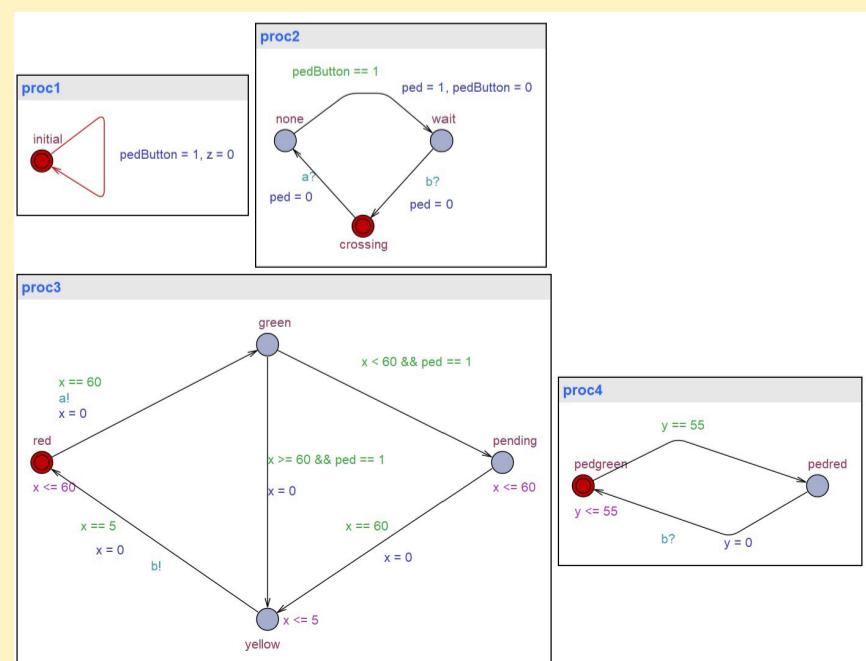
E<> Viking1.safe and Viking2.safe and Viking3.safe and Viking4.safe and Viking5.safe

shortest time : 30 mins

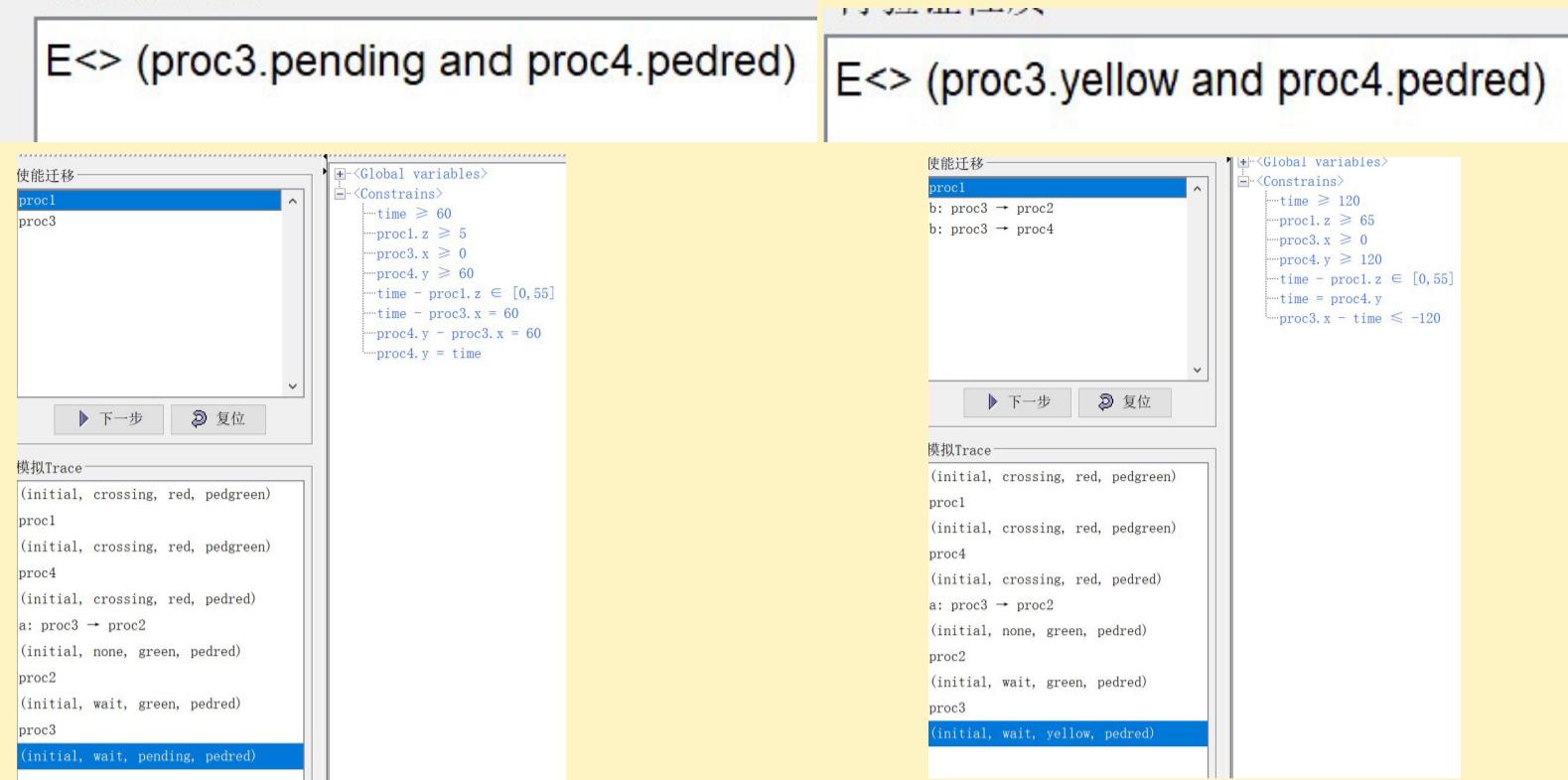
Part B.

- (1) clock variables : clock time (global)
 clock x (local)
 clock y (local)
 clock z (local)

communication channels: chan a
 chan b



test :



the shortest time is 60 seconds

the shortest time is 120 seconds.

(2).

(a). violate, because traffic and crosswalk lights both green are unreasonable. which will make a chaos for vehicles and pedestrians .

E<> (proc3.green and proc4.pedgreen)

result : unsatisfied

(b) violate, in fact. if nobody push pedButton, the traffic light will always be green.

A[] (proc3.green imply proc3.x ≤ 90) result: unsatisfied

(c) satisfy, pending state means pedButton = 1 (pedestrian occur), so max clock is 60 seconds.

A[] (proc3.pending imply proc3.x ≤ 90) result: satisfied

(d) violate , when the crosswalk light is red , the traffic light might be green, pending or yellow.

A[] (proc4.pedred imply proc3.green) result: unsatisfied

(e) violate . yellow only can remain for 5 seconds.

E<> (proc3.yellow and proc3.x = 20) result: unsatisfied

性质列表

E \diamond (proc3.yellow and proc3.x == 20)
 A[] (proc4.pedred imply proc3.green)
 A[] (proc3.pending imply proc3.x <= 90)
 A[] (proc3.green imply proc3.x <= 90)
 E \diamond (proc3.green and proc4.pedgreen)

(3).

(a). E \diamond (proc1.crossing and proc3.pedred), result: satisfied
 $\text{pedCrossTime} = 60$,

When pedestrian is in crossing state and cross-walk light is red. the property is satisfied when $\text{pedCrossTime} = 60$,
 In reality, the condition is disallowed, which is very dangerous.

(b) E \diamond (proc1.crossing and proc3.pedred);

when $\text{pedCrossTime} \geq 55$, bad condition occur, result: satisfied

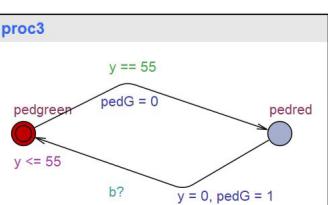
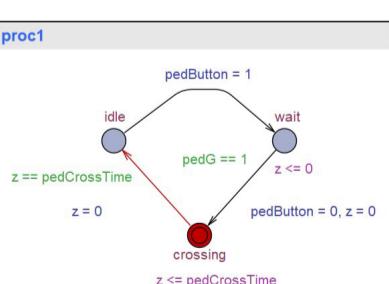
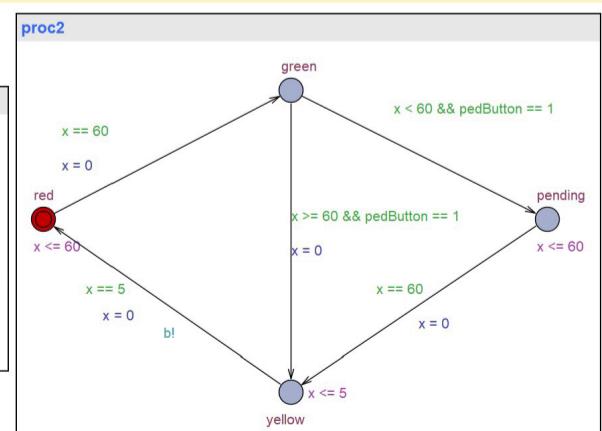
when $\text{pedCrossTime} < 55$, result: unsatisfied, which means no bad condition.

(c) largest safe integer safe value for $\text{pedCrossTime} = 54$.

E \diamond (proc1.crossing and proc3.pedred);

$\text{pedCrossTime} = 54$, result: unsatisfied

$\text{pedCrossTime} = 55$, result: satisfied -



E \diamond (proc1.crossing and proc3.pedred)