Dipartimento di Elettronica, Informazione e Bioingegneria
Politecnico di Milano

Prof. Elisabetta Di Nitto

20133 Milano (Italia)
Piazza Leonardo da Vinci, 32
Tel. (39) 02-2399.3400
Fax (39) 02-2399.3411

Software Engineering for HPC – Written Exam

12th of July 2024

Last name, first name and Id number (Matricola or person code)

Number of paper sheets you are submitting as part of the exam

Rules for the exam

A. Remember to write your name and Id number on each piece of paper that you hand in.
B. You may use a pencil.
C. Incomprehensible handwriting is equivalent to not providing an answer.
D. The exam is open book. You can use your notes, the course slides and additional papers or documents you have collected.
E. You can use the material in printed or in electronic form. In this last case, you can use a plain ebook reader not connected to the internet.
F. You can also use a plain calculator.
G. Read carefully all points in the text!
H. The exam score is up to 16 points. The minimum score to pass the exam is 8 points
I. Total available time at the exam: 1h and 30 mins

Symbolic execution (6 points)

Consider the following function (remember that a % b returns the remainder of the division between a and b):

```
1   int fun(int x, int y)
2   {
3       int z, k;
4       if (x*y == 0)
5           return -1;
6       if (x < 0)
7           x = -x;
8       if (y < 0)
9           y = -y;
10      z = x;
11      k = y;
12      while (x!=y)
13          if(x > y)
14              x = x - y;
15          else y = y - x;
16      if (z % x == 0)
17          return z;
18      else return k;
19  }
```

A. Consider the following two paths:
   Path 1: 1, 2, 3, 4, 6, 7, 8, 10, 11, 12, 13, 14, 12, 13, 14, 12, 13, 15, 12, 16, 17
   Path 2: 1, 2, 3, 4, 6, 7, 8, 9, 10, 11, 12, 16, 18
   For each path do the following:
   1. Use symbolic execution to define the path condition that leads to the execution of the path. Discuss whether the path condition is satisfiable.
   2. If the path condition can be satisfied, define a corresponding test case for the function.

Solution

Symbolic execution for path 1

#	x	y	z	k	Path cond
1, 2, 3	X	Y			
4	X	Y			X*Y!=0
6	X	Y			X < 0 and Y != 0
7	-X	Y			X < 0 and Y != 0
8	-X	Y			X < 0 and Y > 0
10	-X	Y	-X		X < 0 and Y > 0
11	-X	Y	-X	Y	X < 0 and Y > 0
12	-X	Y	-X	Y	X < 0 and Y > 0 and -X != Y
13	-X	Y	-X	Y	X < 0 and Y > 0 and -X > Y
14	-X-Y	Y	-X	Y	X < 0 and Y > 0 and -X > Y
12	-X-Y	Y	-X	Y	X < 0 and Y > 0 and -X > Y and -X-Y!=Y
13	-X-Y	Y	-X	Y	X < 0 and Y > 0 and -X > Y and -X > 2Y

14	-X-2Y	Y	-X	Y	X < 0 and Y > 0 and -X > 2Y
12	-X-2Y	Y	-X	Y	X < 0 and Y > 0 and -X > 2Y and -X-2Y!=Y
13	-X-2Y	Y	-X	Y	X < 0 and Y > 0 and -X > 2Y and -X<3Y
15	-X-2Y	3Y+X	-X	Y	X < 0 and Y > 0 and -X > 2Y and -X<3Y
12	-X-2Y	3Y+X	-X	Y	X < 0 and Y > 0 and -X > 2Y and -X<3Y and -2X == 5Y
16	-X-2Y	3Y+X	-X	Y	X < 0 and Y > 0 and -2X == 5Y and -X% (-X/5) == 0
17	-X-2Y	3Y+X	-X	Y	X < 0 and Y > 0 and -2X == 5Y

The path condition is consistent. Possible test case: x = -10 and y = 4

Symbolic execution for path 2

#	**x**	**y**	**z**	**k**	**Path cond**
1, 2, 3	X	Y			
4	X	Y			X*Y!=0
6	X	Y			X < 0 and Y != 0
7	-X	Y			X < 0 and Y != 0
8	-X	Y			X < 0 and Y < 0
9	-X	-Y			X < 0 and Y < 0
10	-X	-Y	-X		X < 0 and Y < 0
11	-X	-Y	-X	-Y	X < 0 and Y < 0
12	-X	-Y	-X	-Y	X < 0 and Y < 0 and -X == -Y
16	-X	-Y	-X	-Y	X < 0 and -X == -Y and -X % (-X) != 0
18					

The path condition is not feasible as it is not possible that -X % (-X) != 0. Therefore, the path cannot be executed.

Petri nets (10 points)

Consider a traffic semaphore with the usual colors red, green and yellow. The semaphore can be of one color at a time and, in a cyclic manner, after being red, it can become green, then yellow, and then red again to restart the cycle.

1. Assuming that the semaphore colors are modeled in a Petri net through the places R, G and Y, complete such Petri net to model the behavior of the semaphore. Provide a short and clear description of your model.
2. Consider two semaphores synchronized to control a street intersection. Initially, one of them is red and the other green. Moreover, one can move from red to green only when the other moves from yellow to red. Extend the single semaphore Petri net to model the two semaphores and the defined rule. Provide a short and clear description of your choices.
3. Demonstrate that, given an initial marking representing in your Petri net one green semaphore and one red semaphore, it can never happen that the two semaphores are green at the same time.
4. Reflect on the behavior of your Petri net. Which one of the following three sentences is true in your case? Provide a justification for your answer.
    a. One semaphore moves from red to green **at the same time** in which the other moves from green to yellow;
    b. One semaphore moves from red to green **after** the other moves from green to yellow;
    c. One semaphore moves from red to green **before** the other moves from green to yellow;

Solution

Point 1: See Figure 1.

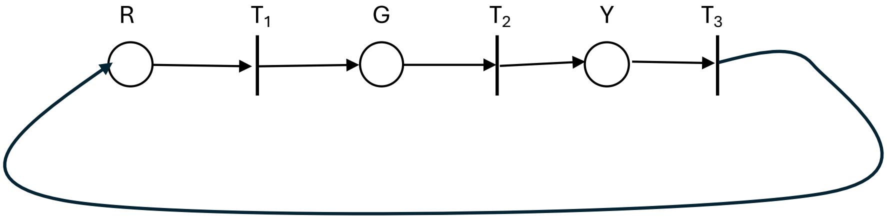

Figure 1. Petri net representing a single semaphore.

Point 2: We can envisage two possible approaches. In Figure 2 the transitions from yellow to red of one semaphore and from red to green of the other occur at the same time. Places R_1, G_1, and Y_1 correspond to the states of the first semaphore, while places R_2, G_2, and Y_2 correspond to the states of the second semaphore. Transitions T1 and T3 are the ones that determine the movements from Yx and Ry to Rx and Gy, while T2 and T4 regulate the transition from G to Y of an individual semaphore independently from the other.
Figure 3 presents a different model that aligns better to real life situations, in which the transition from red to green occurs only a few seconds after the transition from yellow to red of the other semaphore. In this case, Check1 and Check2 are places that receive a token as a result of the transition from Y to R of one semaphore and, thus, enable in the following step, the transition from R to G of the other semaphore.

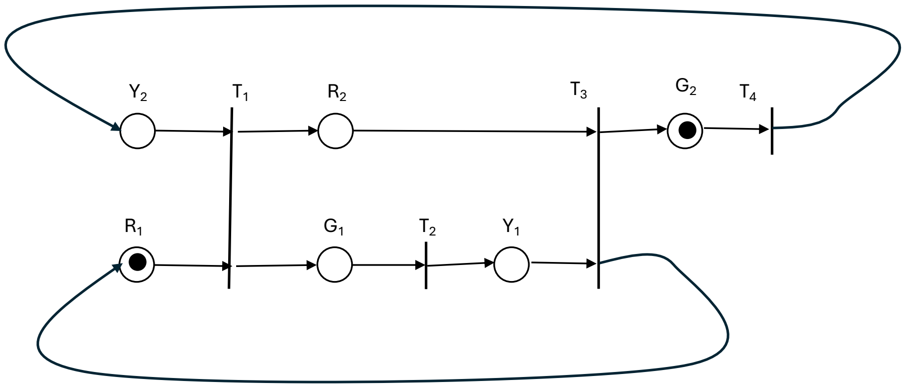

Figure 2. Synchronized semaphores model 1.

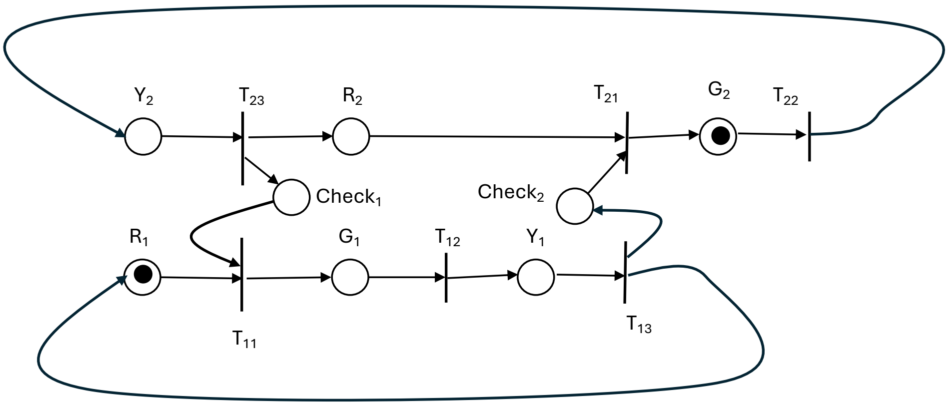

Figure 3. Synchronized semaphore model 2.

Point 3: Figures 4 and 5 show the reachability graphs associated, respectively, to the Petri nets of Figures 2 and 3. Both graphs show that, given the initial marking, the case G1, G2 never occurs, thus demonstrating that the two models fulfill the required constraint.

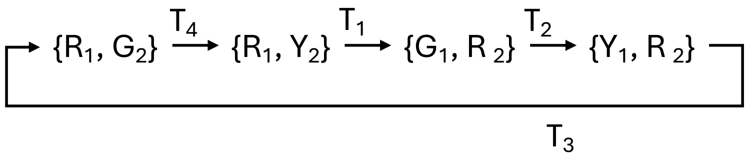

Figure 4. Reachability graph of model 1.

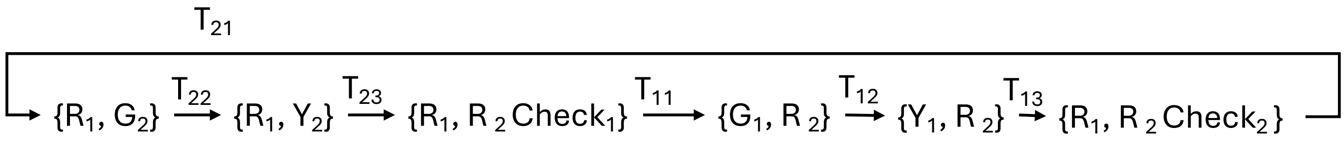

Figure 5. Reachability graph of model 2.

Point 4:
Analyzing the reachability graph of Figure 4, considering as initial marking $\{R_1, G2\}$, we can notice that the transition that causes R_1 to lose its token and G_1 to gain it is T_1 while the transition that causes G_2 to lose its token in favor of Y_2 is T_4. We can also see that T_1 occurs always after T_4. Similarly, the transitions causing a movement of tokens from R_2 to G_2 and from G_1 to Y_1 are T_3 and T_2, respectively. Again, T3 occurs always after T_2. This allows us to conclude that the correct sentence is the following:

- One semaphore moves from red to green **after** the other moves from green to yellow.

We can obtain the same conclusion analyzing the reachability graph in Figure 5. In this case, the relevant transitions are:

- T_{11} for the first semaphore to move from R_1 to G_1 and T_{22} for the second semaphore to move from G_2 to Y_2;
- T_{21} for the second semaphore to move from R_2 to G_2 and T_{12} for the first one to move from G_1 to Y_1.

Given the initial marking $\{R_1, G_2\}$, T_{11} occurs always after T_{22} and T_{21} always after T_{12}.

Side note: the main difference between the two models is in the fact that in the first case the following sentence is true:

- One semaphore moves from red to green **at the same time** in which the other moves from yellow to red,

while in the second model this other sentence is true:

- One semaphore moves from red to green **after** the other moves from yellow to red.