Philipps Universität Marburg

Philipps-Universität Marburg
Fachbereich Mathematik und Informatik

Issue: May 8, 2024
Submission: May 15, 2024

Prof. Gabriele Taentzer
Alexander Lauer

Formal Methods
Homework 2

Algorithm 1: An algorithm that calculates the sum of the first n square numbers.

```
result = 0 ;
tmp = 0 ;
n ∈ {3, 4};
if n == 3 then
    while n > 0 do
        result = n*n + temp ;
        tmp = result ;
        n = n-1 ;
    end
else
    result = (n* (n+1)*(2n+1))/6 ;
end
return result;
```

In this homework we will use the programming language **NanoPromela** in combination with the model checking tool **SPIN**. Download and install the tool. The website spin-root.com provides installation instructions for Windows, Unix/Linux and Mac systems. There are also several tutorials on how to use Spin.

1 Program graphs and NanoPromela (4 points)

Algorithm 1 calculates the sum of the first n square numbers in two different ways and then outputs the result. The term $n \in \{3, 4\}$ means that n takes the value 3 or 4.

a) Program the algorithm in **NanoPromela**. Modify the algorithm so that n is selected from the set $\{3, \dots, 10\}$ and output the result of the calculation to the console using the function `printf()`. Execute your code with SPIN and submit the `.pml` file and a screenshot of the console output. [4 points]

*) Create a program graph that models this algorithm.

*) Calculate for $n \in \{3, 4\}$ and for $n \in \{3, \dots, 10\}$ how many states the transition system of the programme graph has. To do this, consider which values the variables `result` and `tmp` can take.

Philipps Universität Marburg

Philipps-Universität Marburg
Fachbereich Mathematik und Informatik

Prof. Gabriele Taentzer
Alexander Lauer

Formal Methods
Homework 2

Issue: May 8, 2024
Submission: May 15, 2024

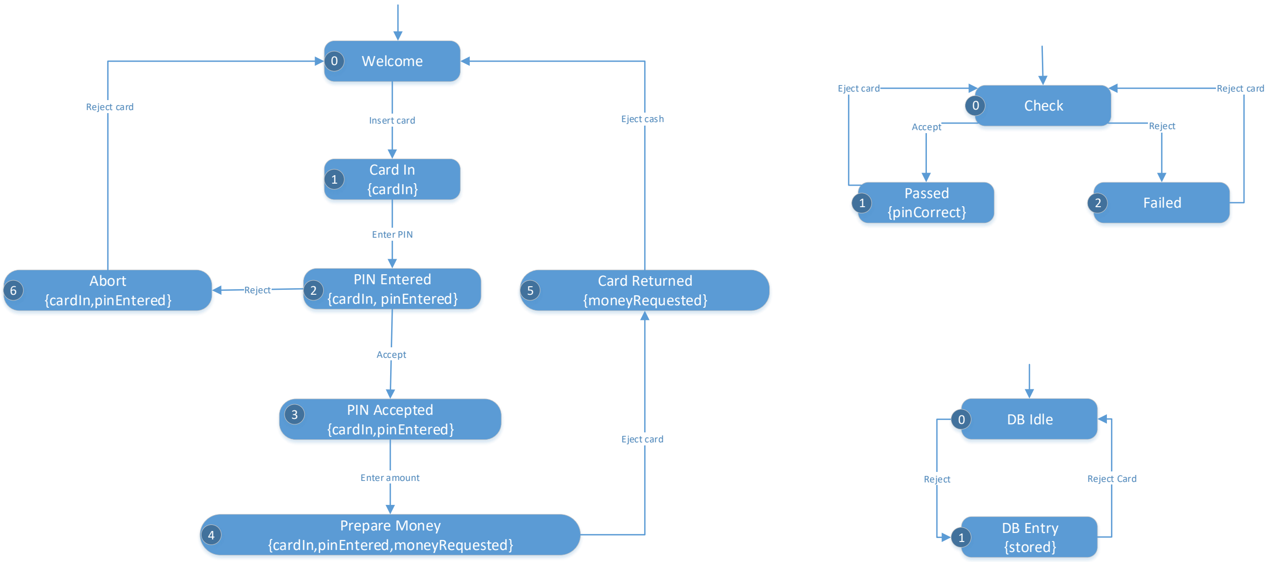

Figure 1: The individual systems of the ATM.

2 Channel systems with NanoPromela (12 points)

In this exercise we consider the ATM from homework sheet 1. The transition systems of the individual components are shown in Figure 1 and the overall system in Figure 2.

a) Program the transition systems shown in Figure 1 in **NanoPromela** as processes. The name of the state and its atomic properties should be displayed on the console for each state (e.g. “PIN Entered: cardIn, pinEntered” when the frontend is in the PIN Entered state). [6 points]

b) Extend your code with a channel system to enable communication between the individual processes. The transition systems should communicate with each other in such a way that together they maintain the functionality of the overall system. [6 points]

   Note: Use a channel of size 0 for synchronous communication between two processes. If the command `c?var` is used, the process will wait at this point until a message has been sent on the channel `c`.

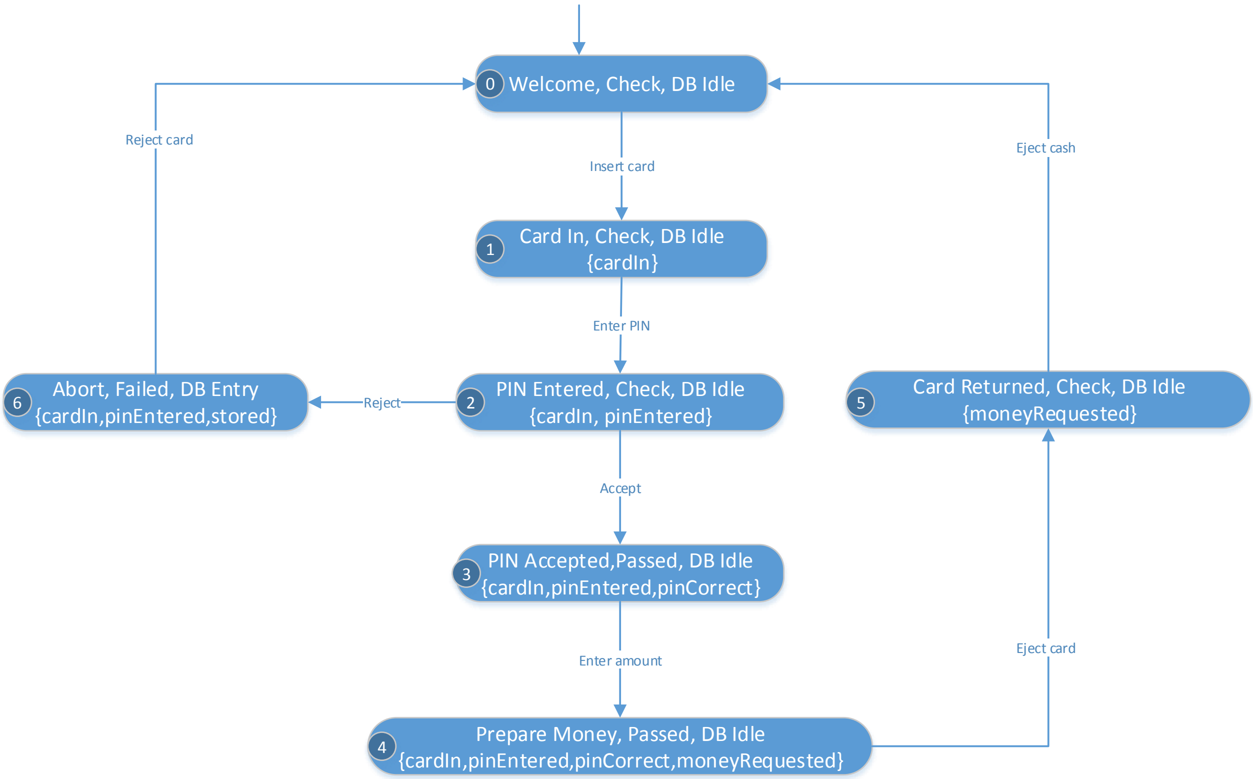
0 Welcome, Check, DB Idle
Reject card
Eject cash
Insert card
1 Card In, Check, DB Idle {cardIn}
Enter PIN
6 Abort, Failed, DB Entry {cardIn,pinEntered,stored}
Reject
2 PIN Entered, Check, DB Idle {cardIn, pinEntered}
5 Card Returned, Check, DB Idle {moneyRequested}
Accept
3 PIN Accepted,Passed, DB Idle {cardIn,pinEntered,pinCorrect}
Enter amount
Eject card
4 Prepare Money, Passed, DB Idle {cardIn,pinEntered,pinCorrect,moneyRequested}

Figure 2: The overall system.