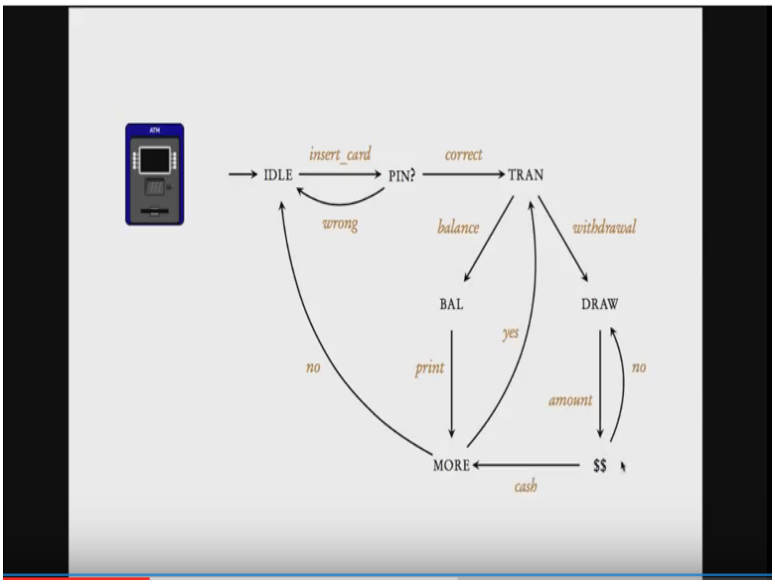
Model Checking
Prof. B. Srivathsan
Department of Computer Science and Engineering
Indian Institute of Technology – Madras

Lecture - 02
Modeling Code Behaviour

In this module, we will see how to model the behavior of a code in succinct way. We will understand this through examples.

(Refer Slide Time: 00:14)



Let us start with simple example of an ATM. The ATM has a controlling code that determines how the ATM should work. Let us try to model it. Initially when the ATM is idle there is no one using it the code is idle as well. When a user inserts a card the code receives the input saying that a card has been inserted. Now the code jumps to a phase where it is waiting for a pin. Now once the user enters the pin.

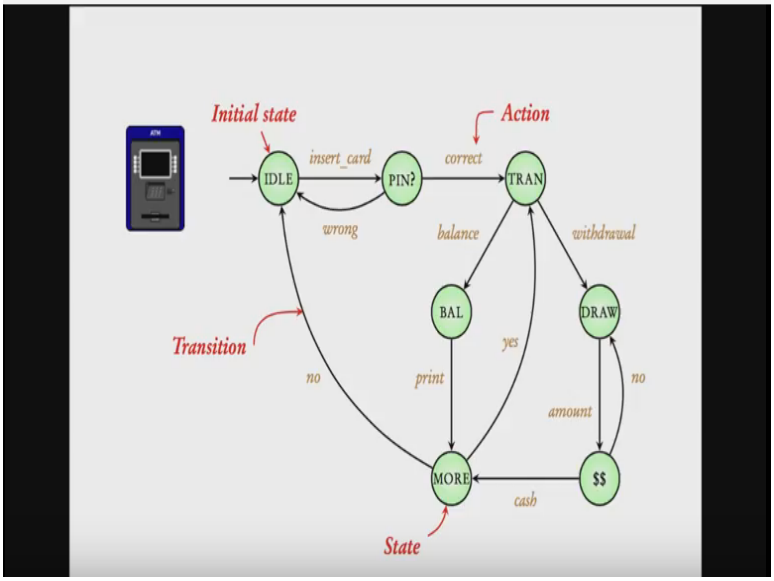
The code determines whether the pin is correct or wrong. If the pin is wrong it goes back to the idle state and after ejecting the card. On the other hand if the pin is correct it goes into a phase where it is looking forward to a transaction by the user. Now let us take only two simple transactions either checking for balance or withdrawal.

These are the two inputs that the user can give. Now if the user has asked for the balance the code moves to a phase where it has process something and give out the balance corresponding to that card and that is what it does. So it prints the balance now it gets into a phase where its asking if the user wants to perform more transactions. If it yes, the code goes back to this transaction phase. If the user says no, the code goes back to the idle state after ejecting the card.

Now let us look what happens if the user has given a withdrawal transaction. So it is come into a phase where it has to process this withdrawal input it will ask for the amount. And once the amount is entered it will deliver cash. If it is possible or if for some reason it cannot deliver the cash it will go back to this state and ask for some other amount. Now once again when its in the more after giving out the cash. If it is in the more transactions phase it will ask the user wants to perform more transactions.

If it is yes it will go here if it is no it will go here people familiar with finite state machines would have recognized that these phases or nothing but.

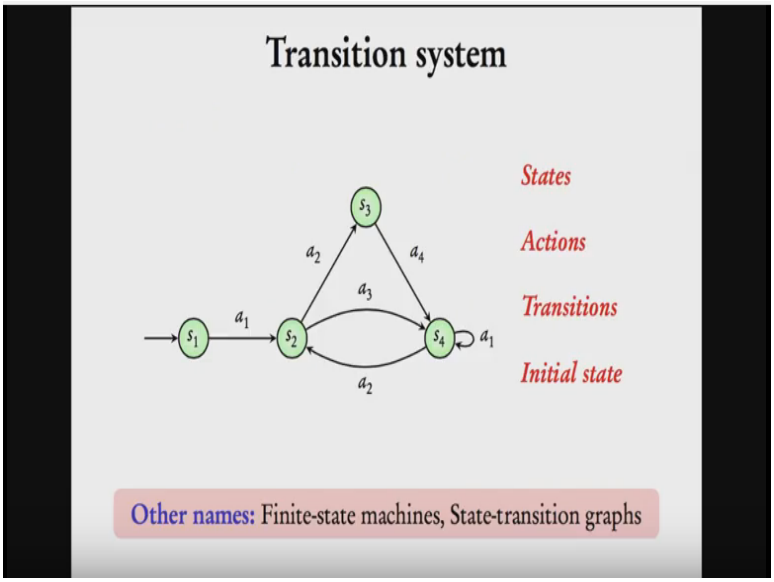
(Refer Slide Time: 03:10)



Different states of the code okay. So this is how we can succinctly represent the behavior of a simple controller of an ATM. These phases are called states. These edges from one phase to the other are called transitions. Each transition occurs due to an action. The action could be external, for example insert card is an external action which is given by the user or the action could be internal print is an internal action okay.

And there is one state which is designated as the initial state so this is how you can represent the behavior of the code contained inside an ATM. This kind of diagram consisting of states actions and transitions is called a transition system.

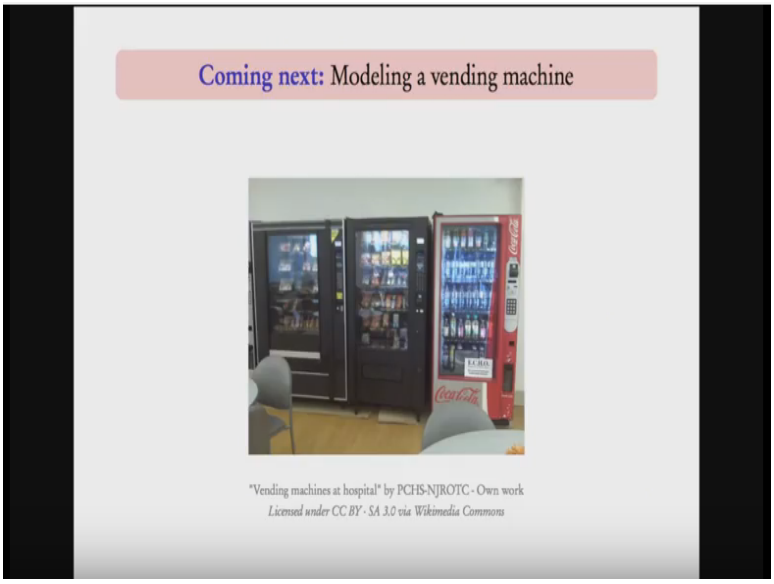
(Refer Slide Time: 04:22)



There are other names toward finite state machines state transition graphs etc. We will be using the word transition system in this course. Let me repeat there are four components to a transition system states. In this transition system, there are states s1 s2 s3 s4 actions here the actions are a1 a2 a3 a4. There are transition to the transition is s1 on an a1 going to s2 is a transition the state s2 on an action a2 going to s3 denoted by this edge is yet another transition and so on.

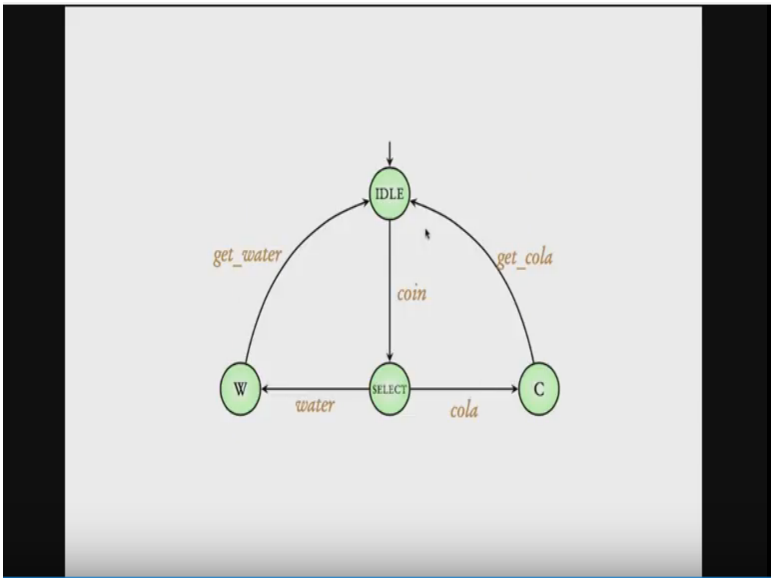
And there is one designated initial state in this diagram. The initial state is s1. So I hope the notion of a transition system is clear.

(Refer Slide Time: 05:20)



Let us look at one more example. Now we will try to model the controller inside a vending machine. So you might have seen such machines in airports and malls etc where you can insert a coin and then select whatever beverage you want and it will give you back the required beverage. Let us try to model the controller present inside such vending machines.

(Refer Slide Time: 05:52)



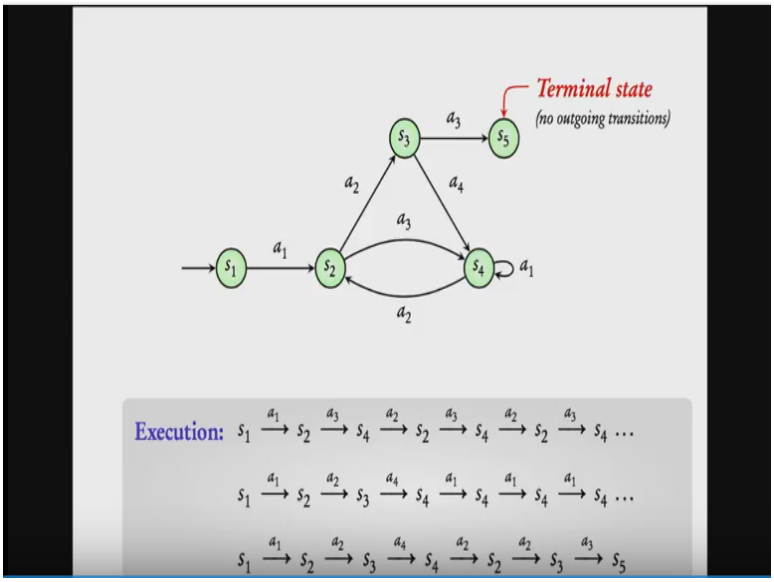
As usual when there is no use the code is idle. When a coin is inserted the code knows that it has go into a phase where the user will select what he wants. Lets assume that there are only 2 kinds of beverages in the machine either water or cola. Now depending on whatever the action is the code goes from the select phase to w or to c and once its in the phase which remembers the selection it will eject the required beverage so this is the simple representation of the code.

There are more complicated details which have been given and we will see later during the course how incorporate the details. And as you see this is succinct representation of the behavior of the code inside a beverage vending machine.

(Refer Slide Time: 06:56)

Now we have seen two examples of transition systems. Let us define some terminology.

(Refer Slide Time: 07:07)

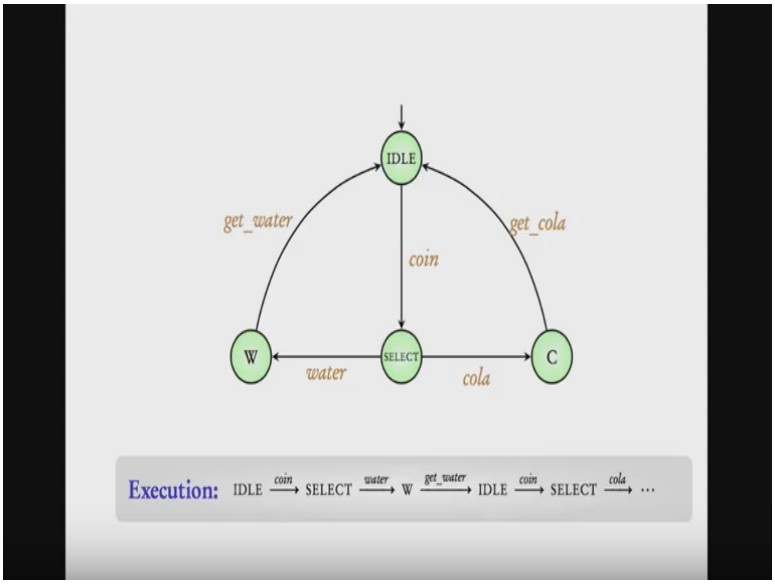


So if in a transition system there is a state with no outgoing transition such a state is called a terminal state just remember this. An execution of the transition system is a sequence of transitions which is either infinite or it is finite and ends in a terminal state. Let us look at these

example executions s1 going to s2, s4, s2, s4, s2, s4 and so on. So one of this execution represents the behavior that starts from s1 goes to s2 then keeps looping around between s2 and s4.

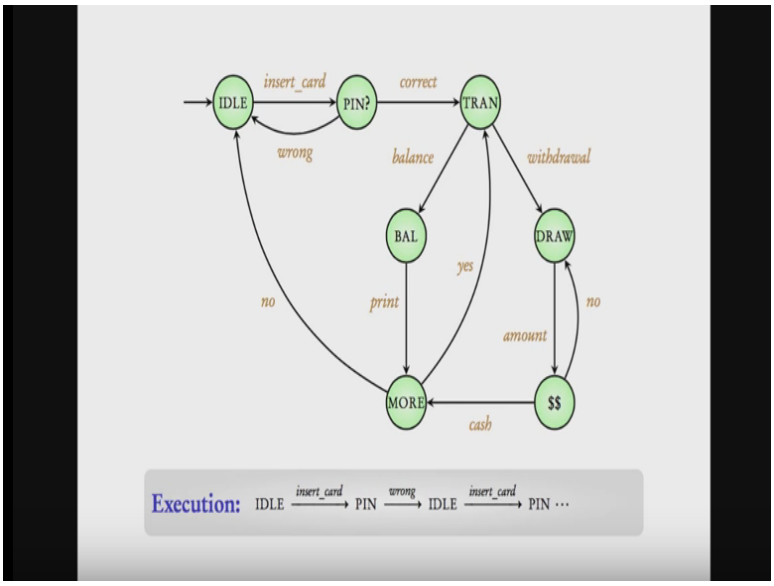
What is this the second execution is s1 s2, s3, s4, s4, s4 and so on this execution represents the behavior that starts at s1 goes to s2 goes to s3 then comes to s4 and keeps looping in s4. Now the final one is the behavior which is like this s1, s2, s3, s4, s2, s3, s5 and at s5 it cannot continue. So this execution represents s1, s2, s3, s4, s2, s3, s5 okay.

(Refer Slide Time: 08:56)



Let us now look at an example of an execution of the code inside a vending machine idle. Select water idle select cola idle select water and so on. So there is one example of an execution. There are of course other executions for example idle select w idle select w idle select w this is 1 execution. The same idle select cola and this loop is an execution. So executions are ways of looking at behaviors of the transition system.

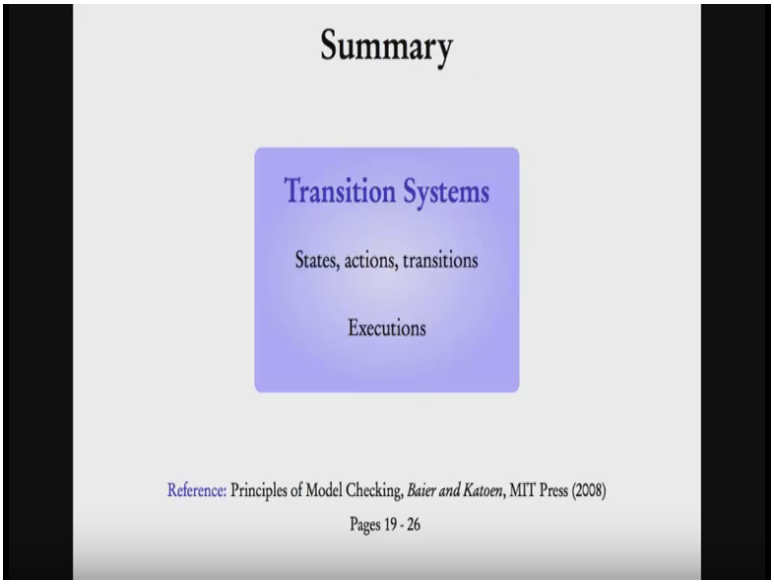
(Refer Slide Time: 09:39)



Now this is the transition system which represents the controller of an ATM. An example of an execution could be insert wrong, insert wrong, insert wrong so on. So this is the scenario when the guy keeps inserting the card and gives out a wrong pin. This could also be other users not necessarily the same user. But this is representing the fact of on the controller is an idle goes to pin, idle, pin and so on.

Get another execution could be idle pin transaction, balance more idle pin transaction, balance more transaction, withdrawal dollar more idle pin idle and so on so these are the behavior of the code. These are the ways in which the code undergoes its transformations.

(Refer Slide Time: 10:43)



So this brings us to the end of the first module. Let me summarize, what we have seen so for. We looked at a way of representing simple code using the notion of transition systems. A transition systems consists of states actions and transitions. It also consists of an initial state we also look that what executions of transition system need. I hope you understand these two concepts and more detailed exposition of these concepts can be found in the book between pages 19 and 26.

However, the exposition is bit more technical and I think you should not worry if you will not get all the details given in the book. As long as, you understand these two concepts the motion of transition systems and the motion of executions in a transition system, I think you are ready to jump to the next module