

CS202 – Subunits 9.2.1-9.2.3

The Saylor Foundation's "Finite State Automata"

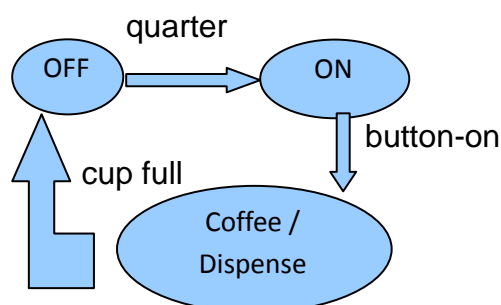
Finite State Automaton, also named finite-state machines, are just a special type of graph. In defining them we will again use set theory, showing the wide utility of sets.

A Finite State Automaton is a tuple (S, I, O, ns, o) , where S is a finite set of state, I is a finite set of input symbols, O is a finite set of output symbols, ns a function from $S \times I$ to S is the next state function, and o is a function from S to O is the output function. There is a distinguished state, s_0 , which is the initial or starting state. An example of a finite state automaton is given in the next paragraph.

Assume a simple vending machine that takes a quarter, button-on, and "cup full" as inputs, and coffee as output. This is a minimal 'no frills' vending machine. We give a graph representation of this machine, its state transition diagram which describes the behavior of the machine.

Finite State Automaton Given by a Transition Diagram

Instruction: For each state we draw a vertex or node. The input function maps a state and an input to a next state. For each such state and input we draw an arc to the node that the input function maps to. The resulting diagram is as follows:



Note that the "/" in the Coffee / Dispense state. The state is written above the "/" and the output is written below. Dispense is short-hand for dispense the cup and the coffee into the cup.

Generating Finite State Automaton from Next-State Tables

A finite state machine can be represented via a graph diagram, as above, or equivalently using a table. Given the diagram one can construct the next state table; conversely, given the next state table one can draw the state diagram.

The next state table for the above diagram is as follows:

Current State	Input	Next State	Output
OFF	quarter	ON	None
ON	button-on	Coffee	Dispense
COFFEE	cup full	OFF	None

We constructed the table by reading the entries from the next-state diagram. For example, in



the diagram, there is an arc that connects the "OFF" state with the "ON" state, labeled "quarter." Hence, we enter "quarter" into the cell having row labeled OFF and column labeled Input.

The reverse process of drawing the transition diagram from a given next-state table is similar, but in reverse. Draw a node for each state in the table; for the above table, the states are the labels of the rows and the labels of the Next State column. Draw an arc for each entry in a cell whose row is a Current State and whose column is Next State. The label on the arc is the entry in the Input column for that Current State and Next State. In each Current State node for which there is an entry in the Output column, write the corresponding entry below the name of the Current State node after a "/."

