

Re: The state machine of no input

Source: <http://sci.tech-archive.net/Archive/sci.math/2004-10/1770.html>

From: Mark \((The WannaBe)\) (mark_hsn_at_hotmail.com)

Date: 10/06/04

Date: Wed, 6 Oct 2004 18:50:33 +0200

----- Original Message -----

From: <briggs@encompasserve.org>

Newsgroups: sci.math

Sent: Wednesday, October 06, 2004 2:12 PM

Subject: Re: The state machine of no input

> In article <[cjv0qb\\$72c\\$1@news.cybercity.dk](mailto:cjv0qb$72c$1@news.cybercity.dk)>, "Mark \((The WannaBe)\)"

<mark_hsn@hotmail.com> writes:

> > Hi

> >

> > I know I am already having some threads going on, but I take my chance,

> > because I suddenly lost my head on this on question. I am still the

Danish

> > wannabe of computer science and that can be the basis for very dumb

> > questions from me, but I try my best not to ask something I could have found

> > out my self by reading a book or searching google. I hope this is the right

> > forum to also ask a question like this and I have been really grateful for

> > previous replies by William Elliot which have helped me to do some cleaning

> > up in my wannabe head full of wrong views of the land of dark magic of

> > computer science.

> >

> > I have been given a state machine, which takes no input, even though it has

> > state transitions with explicit rules for each state transition. I have to

> > prove a property about the language accepted by this state machine, which

> > takes no input. The question says that I should treat the state machine as a

> > DFA. But the state machine takes no input and each state transition is

> > triggered by nothing i.e. they are triggered by the empty string. Now as far

> > as I can figure out, and I am only a wannabe of computer science, then

the

- > > *only language accepted by a state machine taking no input must be the empty*
- > > *string and then only if the starting state is an accepting state.*
- >
- > *How did your textbook define what it means for an automaton to acknowledge*
- > *that an input string is in the language?*
- >
- > *Is it "a string S is in language L defined by automaton A if and only*
- > *if A halts in an accept state on input S "?*
- >
- > *Or is it, perhaps "a string S is in language L defined by automaton A if*
- > *and only if A halts on input S "?*
- >
- > *Either way, I'd say that the language L defined by an automaton A that*
- > *takes no input is either is either all strings or no strings.*
- >
- > *If you layer on the requirement that the automaton actually read*
- > *the entire input then the language L consists of either the*
- > *set containing the empty string or the empty set depending on whether*
- > *the machine accepts or rejects the empty input.*
- >
- > *None of this is substantive. It's all an artifact of details in*
- > *definitions. Much like the question of whether 0 is a natural*
- > *number, in the end it doesn't really change anything.*
- >
- > *John Briggs*

Hi John

Thank you for replying... :-)

My textbook is "Introduction to Automata Theory, Languages, and Computation" by Hopcroft et al. and the problem is about regular languages, so the language L is only accepted if the automaton A halts on input S . In my problem the automaton is initiated in a state, and the clock from then on drives each state transition. Whether or not the automaton can enter an accepting state depends upon which state is chosen as the initial state. Each state transition is well defined but initiated by the clock, but I cannot let a clock be an alphabet or can I? I have to prove whether a state is reachable or not. I believe I can do that, but I am having a problem figuring out for which alphabet this is done? Maybe it is as you say all strings (can enter an accepting state) or no strings (well never reach an accepting state), but the clock drives each state transition? Maybe the clock can be an alphabet; maybe it is just a matter of abstraction? This is really the land of dark magic .

Kind regards

The Danish WannaBe of Computer Science

Mark