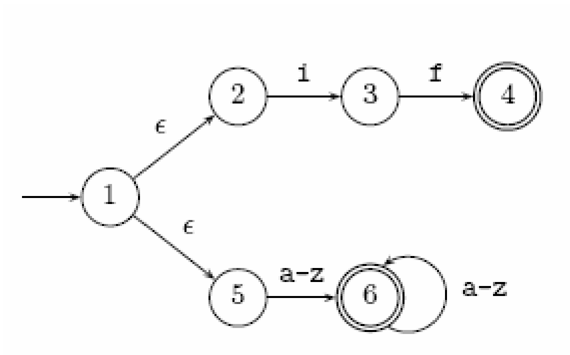
Name_________________

CS503 Homework #3

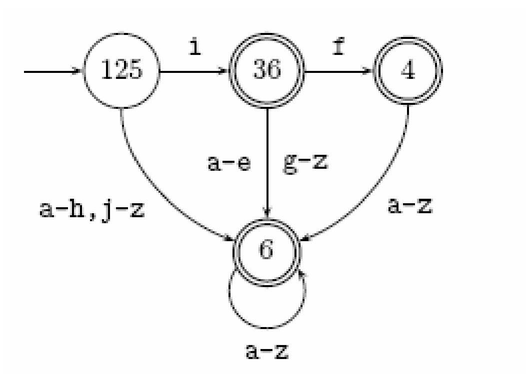
People I talked to, urls I looked at:

#1. Consider the following nfa that will recognize both the keyword "if" and identifiers that consist of at least 1 letter:



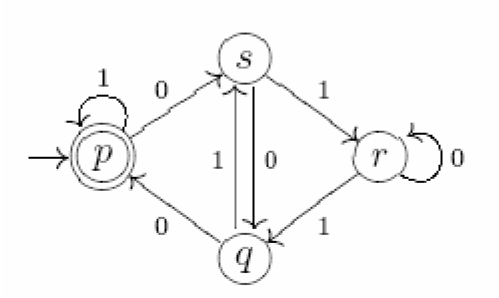
Use the subset construction to convert this nfa to a dfa:

Solution



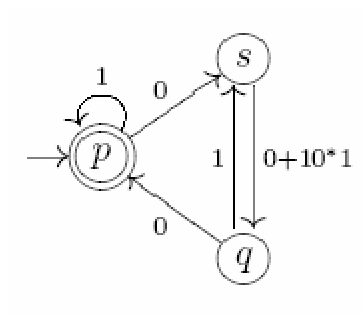
The problem here is that both *f* and *i* are in *a-z* so you need separate entries for f, i, and {a-z} – {f,i} to get a deterministic machine.

#2. Create the regular expression for the following by eliminating states. Please eliminate r first, then s, then q:

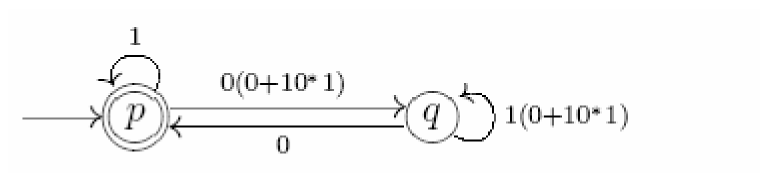


Solution

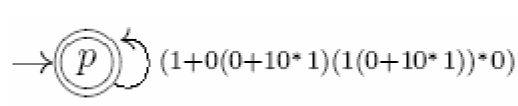
Eliminating r:



Eliminating s:



Eliminating q:



So L(M) = ((1 + 0(0 + 10*1)(1(0 + 10*1))* 0)*

#3. Consider the following operation -3 on regular languages L:

$L^{-3} = \{w \mid y\,w\ \varepsilon\ L \text{ and } |y| = 3\}$

Show regular languages are closed under the -3 operation.

Solution

A regular language L has a fa, M, such that L = ℓ(M).

Add a new start state and λ-transitions to all states that are reachable by a path of length 3 from the original start state of M. This new nfa accepts L^{-3}.

#4. Show that it is decidable whether a regular language, L, contains 1000 strings or more.

If the dfa for L contains a cycle on a path from the initial to final state, then it accepts an infinite number of strings, so certainly accepts 1000 or more.

If there is no cycle from the initial to a final state, just count the number of paths from the initial to the various final states. If there are 1000 or more such paths, L contains 1000 strings or more. If there are fewer, then L does not accept 1000 strings or more.

#5 Use the pumping lemma to show
a) L = {w | w contains twice as many *a*'s as *b*'s} is not regular

Proof

Note that $L \neq \{ a^{2n}b^n \mid n \geq 0\}$!!!

If L were regular, then there is a dfa, M, with *k* states accepting L.

Pick $z = a^{2k}b^k$

Then, since $z \varepsilon L$ and $|z| \geq k$, by the pumping lemma:

$z = u\,v\,w$ with $|u\,v| \leq k$, length(v) >0 and uv^iw is also in L for all $i \geq 0$.

Because $|u\,v| \leq k$, *u v* is all a's and since length(v) >0, $v = a^j$, some j.

When i = 2, we have the string: $u\,v\,v\,w = a^{2k+j}\,b^k$

which has more than twice as many *a*'s as *b*'s. Thus u v v w is not in L which is a contradiction.

Therefore the language is not regular.

b) $L = \{0^n \mid n \text{ is a power of } 2\}$

Proof

If L were regular, then there is a dfa, M, with *k* states accepting L.
Pick $z = 0^m$ where $m = 2^k$

Then, since $|z| = 2^k \geq k$, by the pumping lemma:

$z = uvw$ with $|uv| \leq k$, length(v) >0 and uv^iw is also in L for all $i \geq 0$.

Since $|uv| \leq k$ and length(v) >0, there are between 1 and k 0's in v.

$1 \leq |v| \leq k$

So $2^k + 1 \leq |u\,v\,v\,w| \leq 2^k + k < 2^k + 2^k = 2^{k+1}$

So u v v w has length between $2^k + 1$ and $2^k + k$.

So |uvvw| cannot be a power of 2 and thus uvvw is not in the language.
Therefore L is not regular.