Main Wed Apr 12 7:49 PM

JFLAP : <untitled1>

File Input Test View Convert Help

Editor Multiple Run

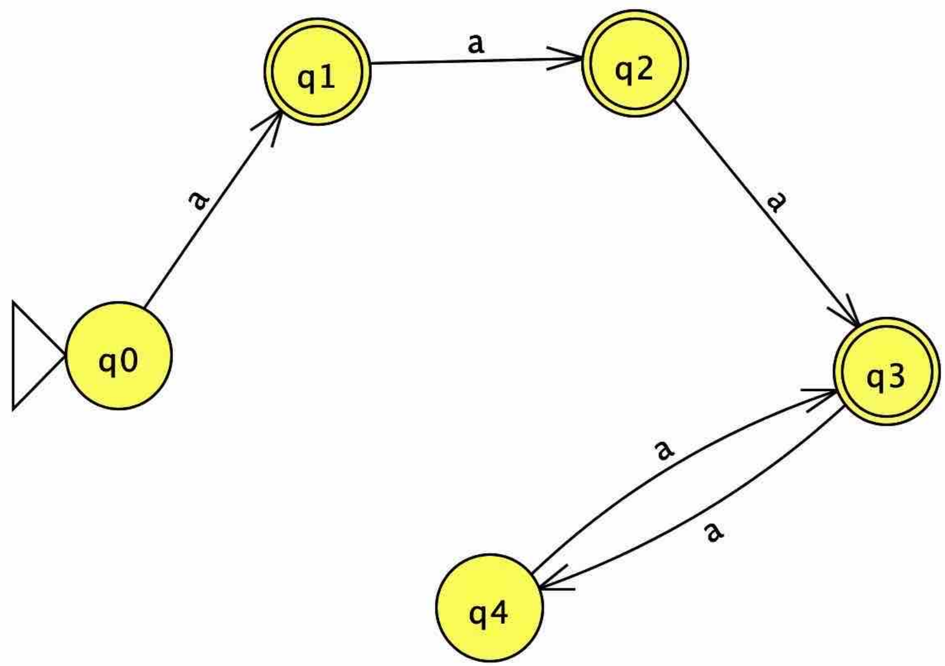


Table Text Size

Input	Result
aa	Accept
aaaaaa	Reject
aaa	Accept
aaaaa	Accept
aaaaaaab	Reject
aaaaaaaa	Reject

Load Inputs Run Inputs Clear Enter Lambda View Trace

Main   Wed Apr 12 7:57 PM

JFLAP : <untitled1>

File Input Test View Convert Help

Editor | Multiple Run

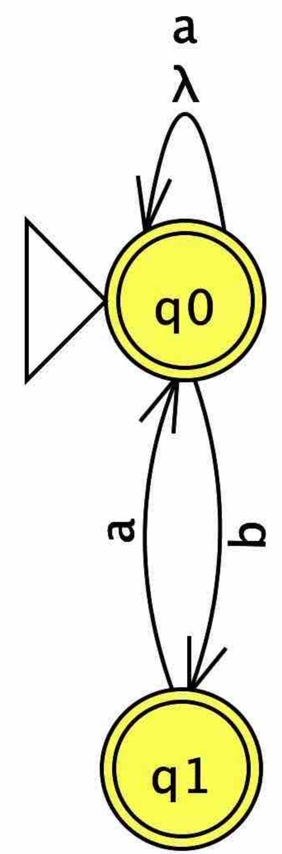


Table Text Size

Input	Result
b	Accept
a	Accept
aa	Accept
babab	Accept
babaa	Accept
ababb	Reject

Load Inputs | Run Inputs | Clear | Enter Lambda | View Trace

Main Wed Apr 12 8:01 PM

JFLAP : <untitled2>

File Input Test View Convert Help

Editor Multiple Run

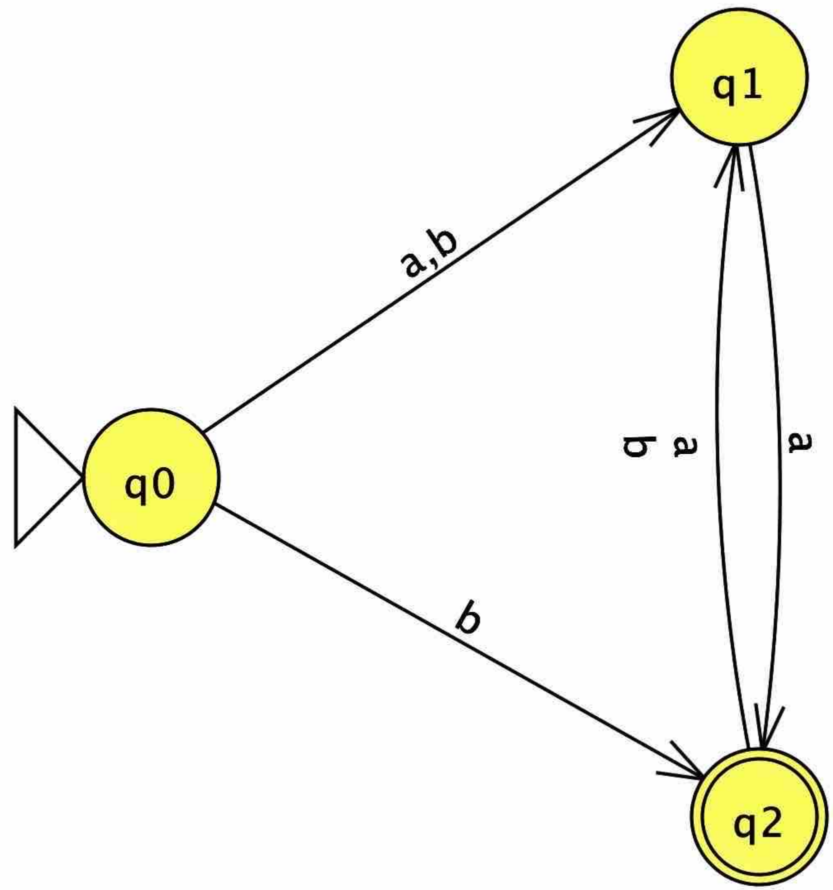
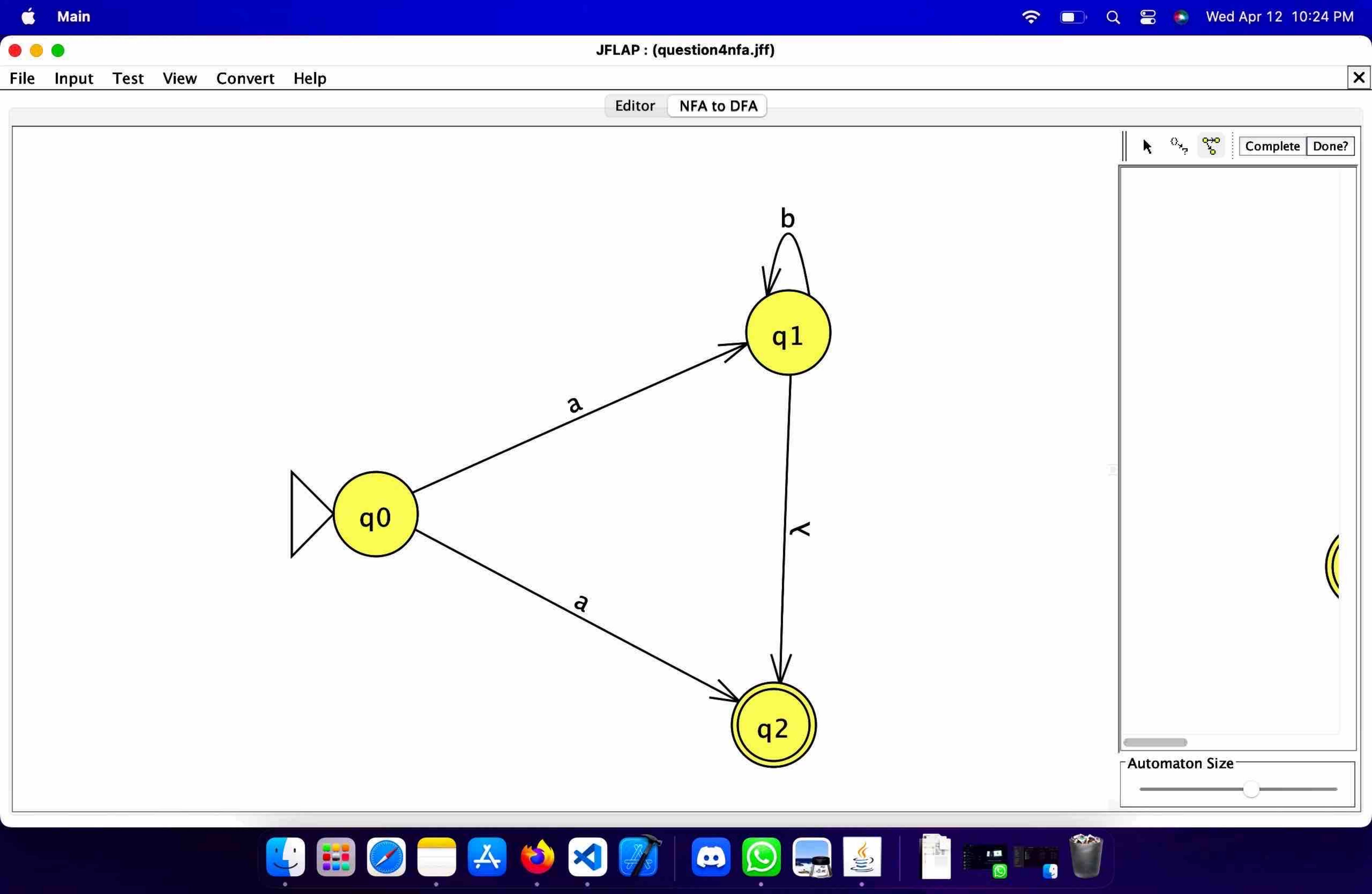


Table Text Size

Input	Result
aa	Reject
ab	Reject
b	Accept
bb	Reject
aaa	Reject
abbb	Reject
aabbbb	Reject
babab	Reject
bbaab	Reject

Load Inputs Run Inputs Clear Enter Lambda View Trace

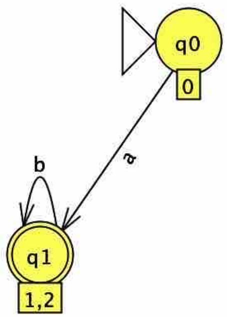

Main
Wed Apr 12 10:24 PM
JFLAP : (question4nfa.jff)
File
Input
Test
View
Convert
Help
Editor
NFA to DFA
Complete
Done?
b
q1
a
q0
λ
a
q2
Automaton Size

Main Wed Apr 12 10:23 PM

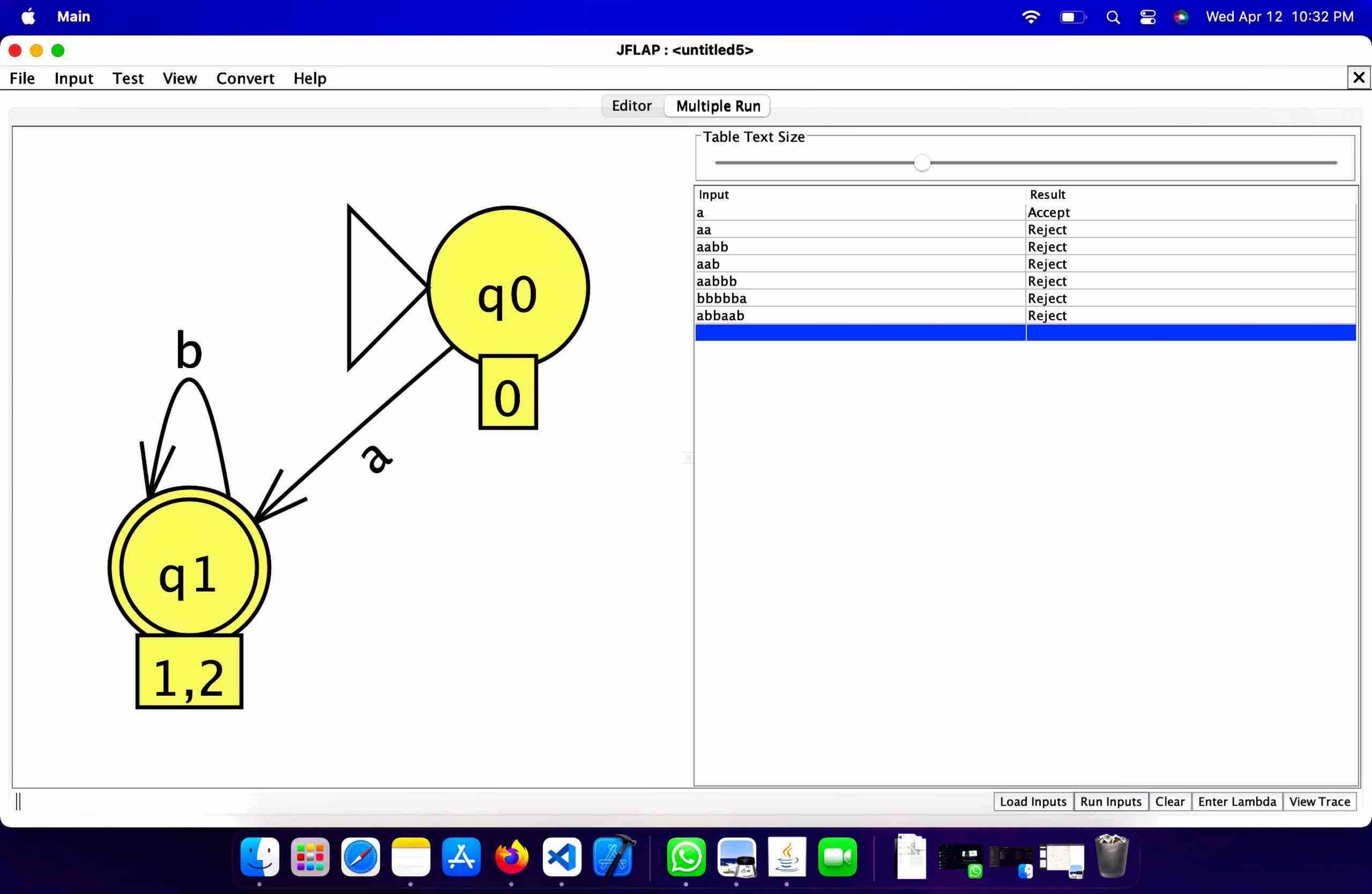
JFLAP : (question4dfa.jff)

File Input Test View Convert Help

Editor



Automaton Size



Main
Wed Apr 12 10:32 PM
JFLAP : <untitled5>
File
Input
Test
View
Convert
Help
Editor
Multiple Run
q0
0
a
b
q1
1,2
Table Text Size
Input Result
a Accept
aa Reject
aabb Reject
aab Reject
aabbb Reject
bbbbba Reject
abbaab Reject
Load Inputs
Run Inputs
Clear
Enter Lambda
View Trace

Q5 If language contains a string $v_1, v_2, v_3, \ldots v_n$.
One possible expression is n.
$v_1 \cup v_2 \cup v_3 \cup \ldots \cup v_n = \bigcup_{i=1}^{n} v_i$

U is commonly written 1, especially on computer
Since there is a regular expression for the language. it is regular.
suppose language L contains of string $a_1, a_2, a_3, \ldots a_n$. Consider the following NFA to accept L:
it has a start state S and accepting state A. In between S and A there are n different path q states one for each a. it can get only from the language of the i'th path to end if sees exactly string a. there are E transition from S to beginning of each path and from the end of each path to A.

Q1 Since L is a regular language we can construct a corresponding dFa N, such that $L(N) = L$.
→ By defn L^R consists of all strings in language L in reverse order. we will construct a NFa, N_R, representing L^R such that $L(N_R) = L^R$.
N_R will contain an addition start state with λ. transition to final state of N. The direction of every transition is reversed. Also the start state of N will be the final state of N_R. The construction of nFa N_R is as follows.

Let $N = (\Phi, \Sigma, \delta, q_n, F)$

$N_R = (Q \cup \{q_0\}, \Sigma, \delta_r, q_r, \{q_n\})$

Set of states of N_R = set of states of N along with $q_0 = Q \cup \{q_0\}$

Σ = alphabet of N_R = same as N

q_r = start state of N_R

$\{q_n\}$ = Set of final state of N_R = start state of N.

transition function : $\delta_r(q, a) = \{q_i : \delta(q, a) = q\}$

$\delta_r(q_r, \lambda) = F$

$\delta_r(q_r, a) = \phi$, if $a \neq \lambda$

Now we can show that $L^R = L(N_R)$
$w \in L^R$. if $w^R \in L$ there is a walk on a transition graph of N with label w^R from q_n to some $q_i \in F$. if there is walk on the transition of N_R from q_r to q_i with label λ and a walk from q_i to q_n with label w iff $w \in L(N_R)$.

Since L^R can be represented by a nFa, it is regular.